AD-A064 800 CONTROL DATA CORP MINNEAPOLIS MINN DIGITAL IMAGE SYS--ETC F/G 8/2
DIGITAL CARTOGRAPHIC STUDY AND BENCHMARK.(U)
DEC 78 D J PANTON, M E MURPHY, D S HANSON DAAG53-75-C-0195
UNCLASSIFIED CDC-76003-78-R6 ETL-0168 NL

1 OF 3
ADA
064800

(12)

SC

CONTROL DATA

LEVEL III

A043156

ETL - 0168

DIGITAL CARTOGRAPHIC STUDY AND BENCHMARK

FINAL REPORT

Prepared for:

U. S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia

Approved for public release; distribution unlimited.

Prepared by:

D. J. Panton
M. E. Murphy
D. S. Hanson

December 1978

Control Data Corporation
DIGITAL IMAGE SYSTEMS DIVISION
2800 East Old Shakopee Road
Minneapolis, Minnesota 55420

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER ETL-0168 ✓	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) DIGITAL CARTOGRAPHIC STUDY AND BENCHMARK.		5. TYPE OF REPORT & PERIOD COVERED FINAL TECHNICAL REPORT, 4 JUN 1975 - OCT 1978,
		6. PERFORMING ORG. REPORT NUMBER CDC-76003-78-R6
7. AUTHOR(s) D. J. Panton M. E. Murphy D. S. Hanson		8. CONTRACT OR GRANT NUMBER(s) DAAG53-75-C-0195
9. PERFORMING ORGANIZATION NAME AND ADDRESS CONTROL DATA CORPORATION 2800 EAST OLD SHAKOPEE ROAD MINNEAPOLIS, MINNESOTA 55420		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS U. S. ARMY ENGINEER TOPOGRAPHIC LABORATORIES (USAETL) FORT BELVOIR, VA. 22060		12. REPORT DATE DEC 1978
		13. NUMBER OF PAGES 274 276 p.
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) SAME		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

SAME

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

DIGITAL IMAGE PROCESSING	PARALLEL PROCESSING
DIGITAL STEREO MAPPING	DISTRIBUTIVE COMPUTING
ANALYTICAL PHOTOGRAMMETRY	STEREO COMPILATION
AUTOMATED MAPPING	DIGITAL TERRAIN ELEVATION DATA COLLECTION
IMAGE CORRELATION	IMAGE MATCHING

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

SEE FOLLOWING PAGE

408 732

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

ABSTRACT

A flexible algorithm has been developed to meet the changing requirements for generating terrain data from digital stereo sensor records. The algorithm includes an image matching procedure in which parallax components are determined by automatically correlating conjugate image features. The algorithm is adaptive and can handle various types of sensor and natural terrain conditions. Reliability monitoring of the output terrain data is performed on the basis of the in-process analysis of local image areas. The reliability measure dictates various strategies that the algorithm can apply in image areas where automatic correlation is difficult.

The algorithm was implemented on a distributive network of parallel digital processors. In this system, production speed is attained because of the inherent parallelism of the modular processors. Flexibility is maintained because the processors are microprogrammable. In this way, new sensor imaging characteristics and new algorithm strategies can be incorporated without disturbing the fundamental software and hardware structure of the system. Production times for compiling a representative stereo model on this parallel configuration far exceed the capability of general-purpose computers.

Based on the benchmark implementation, a production system has been designed which incorporates a full interactive image display capability. This allows an operator to interactively monitor, control, and edit the stereo compilation processes by means of a stereo CRT display, keyboard, and trackballs.

1498 A

FOREWORD

This final report is the collection of the five interim technical reports that correspond to the five completed phases of the Digital Cartographic Study and Benchmark. The collection has been made for consistency's sake, since the significant results of the study have been spread out in time and over five separate interim reports.

The purpose of the Digital Cartographic Study and Benchmark was to develop and evaluate algorithms suitable for stereo mapping in an all-digital environment, to implement a representative algorithm on micro-programmable parallel processors as a benchmark, and to recommend a total system design for a digital stereo mapping and editing system.

The First Interim Technical Report summarizes the early algorithm development that supported the choice of a specific technique for digital stereo mapping. The Second Interim Technical Report describes the process of algorithm logic-level reconstruction which was necessary for adapting the matching algorithm for parallel processing implementation. The Third Interim Technical Report then describes the actual benchmark implementation, timing, and performance. Since the basic techniques and strategies of the algorithm underwent several refinements in the course of the study, the Fourth Interim Technical Report summarizes the complete matching algorithm capability to date. Finally, the Fifth Interim Technical Report contains the specifications and design for a digital stereo mapping and editing system.

ETL - 1068

DIGITAL CARTOGRAPHIC STUDY AND BENCHMARK

FINAL REPORT

Prepared for:

U. S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia

Approved for public release; distribution unlimited.

Prepared by:

D. J. Panton
M. E. Murphy
D. S. Hanson

December 1978

Control Data Corporation
DIGITAL IMAGE SYSTEMS DIVISION
2800 East Old Shakopee Road
Minneapolis, Minnesota 55420

DIGITAL CARTOGRAPHIC STUDY AND BENCHMARK

FIRST INTERIM TECHNICAL REPORT

Prepared for:

U. S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia

Contract DAAG53-75-C-0195

Prepared by:

D. J. Panton
M. E. Murphy

October 1975

Control Data Corporation
DIGITAL IMAGE SYSTEMS DIVISION
2800 East Old Shakopee Road
Minneapolis, Minnesota 55420

ABSTRACT

The key problem in the automatic generation of digital terrain data is the matching of conjugate points on a stereo pair of aerial photographs in an accurate and timely manner. The work described in this report centers around the development of suitable algorithms and systems procedures to perform this matching task in an all-digital environment.

Two approaches to automatic image matching have been investigated; a strip processing approach and a block processing approach. Of the two approaches, it has been found that the block processing approach is more adaptable to the requirements of digital terrain data collection. Therefore, this approach is being investigated further for implementation in an array of fast, microprogrammable processors to provide a benchmark of matching system parameters and performance.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1.0 INTRODUCTION

Automatic systems for the compilation of digital terrain data must be fast, accurate and flexible. In addition, the cost of such systems must be in proportion to the quality of data they generate. The need for automatic cartographic systems is unquestioned, considering the volume of photographic data that needs processing. An important consideration is the optimum technology required to perform this task of cartographic data processing.

This technical report is a summary of the first phase of ongoing work in the development of fast, accurate, and flexible algorithms and systems to produce the desired mapping products.

The primary area of consideration in generating digital terrain data is the automatic matching of conjugate points in a stereo pair of aerial photographs. This is the area in which computational speed is of utmost importance and in which a great deal of algorithm sophistication is required. Clearly, all of the research regarding the optimum approach for such a task is not yet in.

Therefore, two approaches to the problem of image matching which have been used in the past by Control Data for automatic registration and change detection purposes are being investigated. The study described in this report involves the redesign and modification of the two approaches to efficiently and accurately process stereo imagery. The overall goal of the effort is to choose the better approach for implementation in fast, microprogrammable processors as a demonstration and verification of feasibility.

The two image matching approaches, called the strip approach and block approach, have been generally described in a previous technical report [1], and these aspects will not be rediscussed here. Only the modifications, new aspects, and new results will be treated. Suffice it to say that since both approaches utilize the correlation coefficient as the image matching metric, what the study is really comparing are the two philosophies behind the matching approaches.

The strip processing approach is very fast and production-oriented. It is fast because it incorporates a line-by-line error correcting process. That is, the process knows where on the image to move next based on the errors it sees now. On the other hand, the block approach is a slower process, incorporating a rather deterministic predict-ahead mechanism. The study has shown that, because of these differences in approach, differences in the output digital terrain data occur and a number of trade-offs emerge.

2.0 MODIFICATION OF STRIP APPROACH: INTRODUCTION

Throughout this section of the report, we will be addressing the problem of successfully matching stereo imagery with an automatic strip process. This automatic strip processor will be referred to as program TRAK. We will discuss enhancements to TRAK made possible by the nature of stereo imagery. In addition, we will also address problems identified by an earlier attempt to use the strip process for this application. We will discuss solutions implemented for those problems and will indicate results, both visual and statistical, which we were able to achieve.

2.1 INCORPORATION OF EPIPOLAR Y-AXIS CONTROL

Previous applications of the strip process, such as side looking radar, ERTS multispectral data, and high resolution aerial photography, have included a correlation process using five patch sites separated by a distance of from one to seven cells as indicated in Figure 2-1.

The effective size of the five sites is some patch height (user defined) by some patch width. A weighted accumulation of the required vector dot products provides an effective patch width dimension thereby reducing required memory for the actual inner products necessary to compute correlation. Using these five correlations, an interpolated maximum point on the correlation surface is found as illustrated in Figure 2-2. Deviations, or spatial error terms, are computed and then incorporated into warp equations as shown in Figure 2-3. Subsequently, these error terms drive the correction process.

For the application of matching stereo imagery, north and south sites were eliminated from the correlation and subsequent warp update process. Epipolar line geometry was incorporated into the process thereby providing y-axis control. X-axis matching is as performed in previous applications.

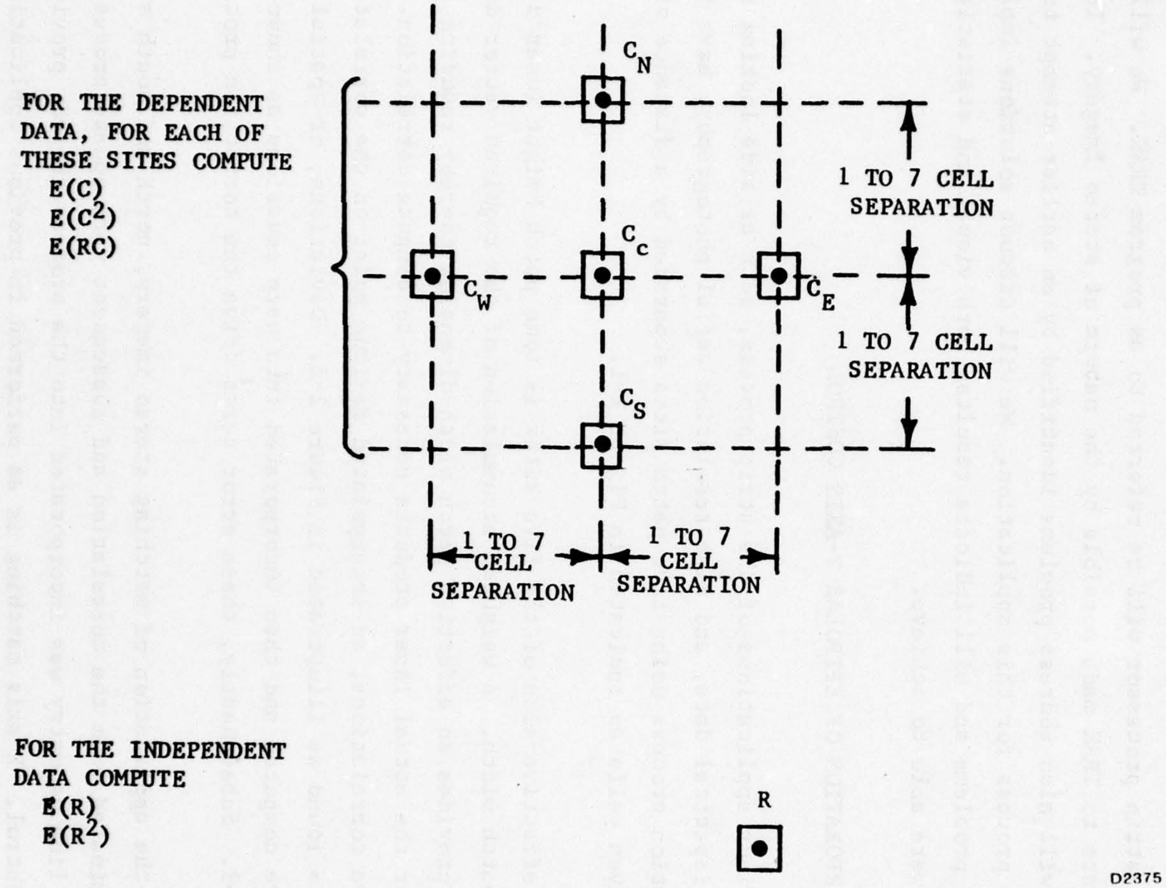


Figure 2-1. Inner Products Requirements Per Strip

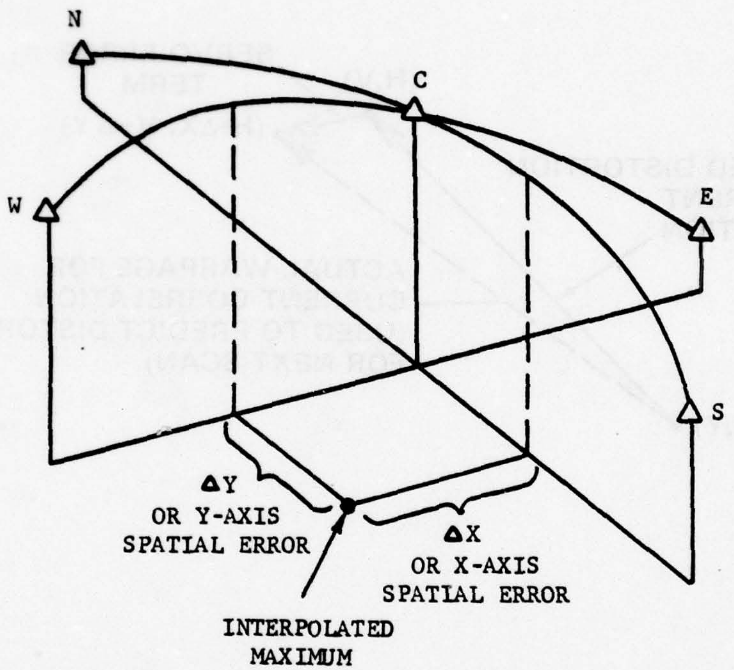


Figure 2-2. Locating Maximum Point On Correlation Surface

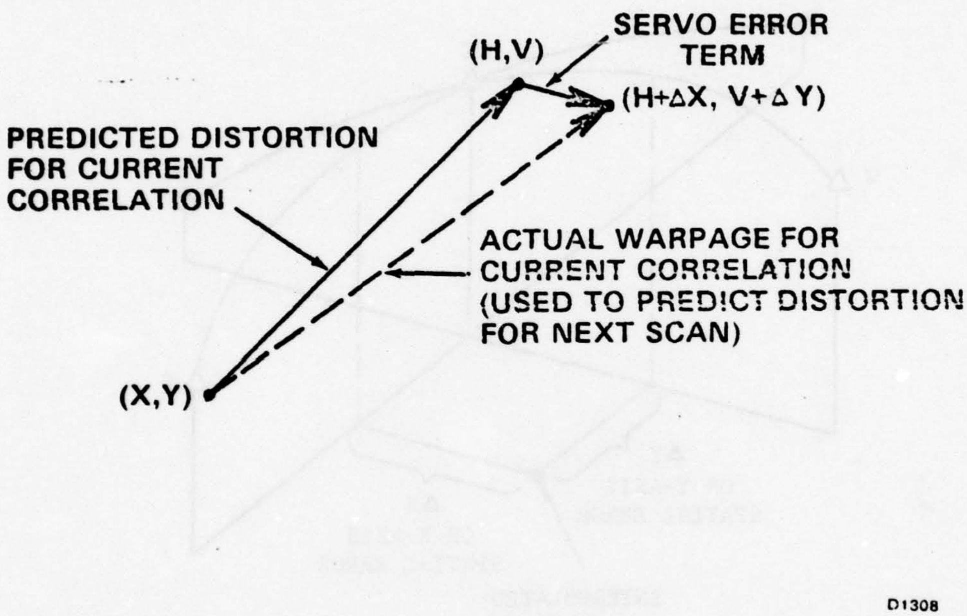


Figure 2-3. Relationship of the Maximum Correlation Coefficient Location as Used to Update Prediction Equation.

2.2 PROBLEMS IDENTIFIED BY PREVIOUS EXPERIENCE

Based on a previous study done by Digital Image Systems Division for USAETL, the following problems were identified in the strip process: First, the problem of effectively modeling the terrain in the creation of a synthetic scan line for subsequent correlation and match update; second, the problem of allowing the warp update process enough freedom to follow terrain, and yet retain enough stability in areas of dissimilarity; and third, the problem of interlocking strip information to effectively guide strips in hard-to-match areas. In the following three sections, we shall address each of these problems and the solutions we implemented for each.

2.2.1 Approximating High Order Equation In Obtaining Synthetic Scan Line

It is well known that successful matching of stereo images requires that x-parallax, or differences in the placement of matching pixels in stereo images due to terrain relief be removed. Since the TRAK program is a line-by-line process, a synthetic scan line from the dependent image of the stereo pair must be created for each line of the independent image. The proper creation of this synthetic scan line is of utmost importance in determining the current parallax signal, which in turn is used to drive the process. The building of this equivalent synthetic line is accomplished by defining positions within a buffer containing a section of the dependent image which are the equivalent of the strip centers in the independent image. This relationship is shown in Figure 2-4. Interpolation between strip centers is linear and uses a nearest neighbor criterion for the actual access of matching pixels. By retaining the linear interpolation technique and moving strip centers closer together, a much better approximation to the matching synthetic scan line, which is really a function of the terrain being imaged, can be obtained. This effectively gives TRAK the capacity to closely approximate a warp equation of an order which is entirely dependent on the terrain being encountered. In addition, correlation patch shaping is performed for each line since inner product memories can be updated with information resulting from the previous line's parallax signal.

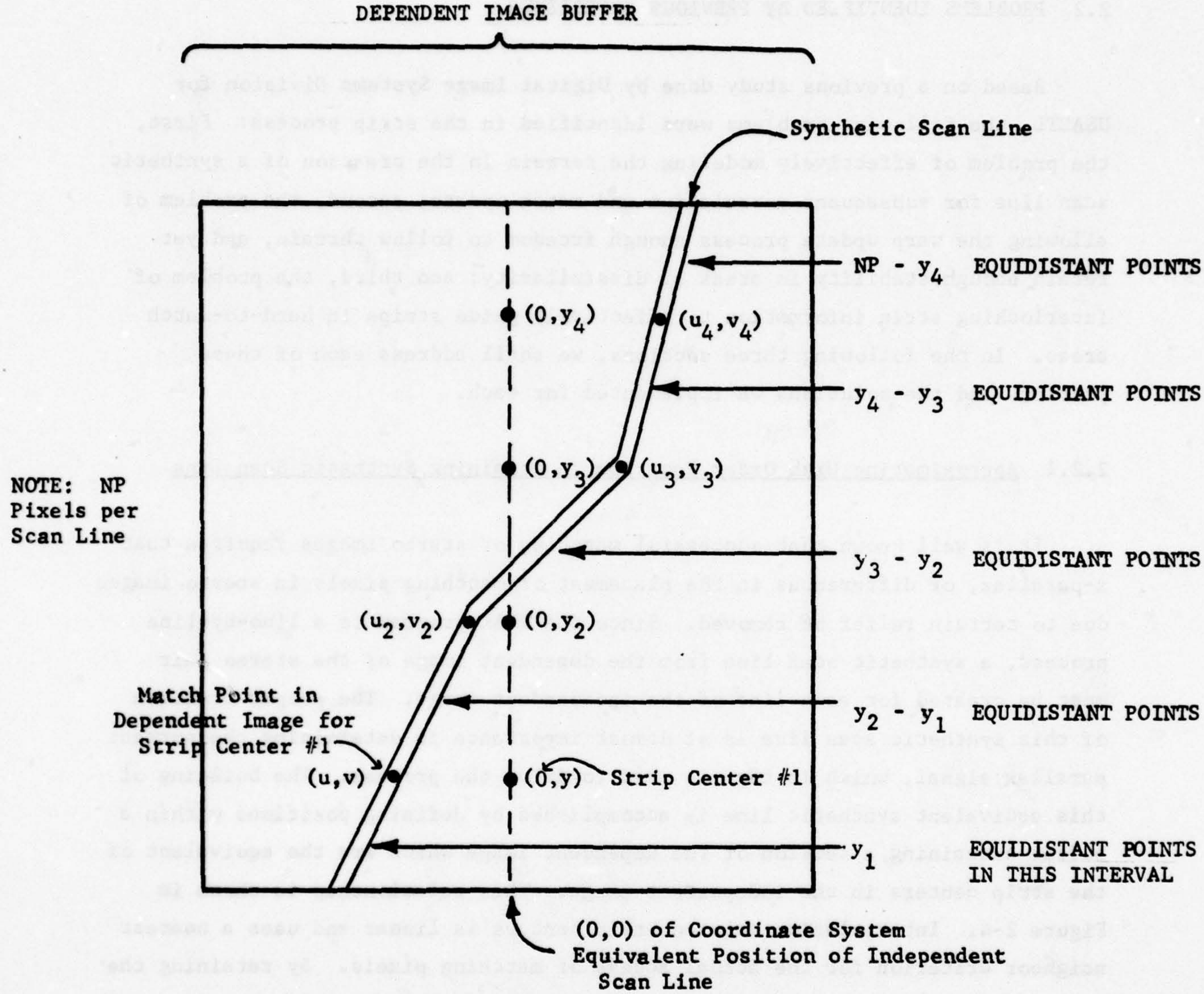


Figure 2-4. Accessing Of Synthetic Scan Line From Dependent Image Buffer Window

D2378

2.2.2 Constraint Versus Flexibility in the Choice of System Damping

System damping in the TRAK program may be defined as the amount of filtering or smoothing needed to sufficiently remove noise effects in the error signal values produced by the correlation process. There are two unique and yet related methods by which the present version of TRAK is damped; the first is the rate at which inner products used for the correlation process are allowed to accumulate (definition of effective patch width), and the second is the rate at which error signals, resulting from location of the interpolated maximum on the correlation surface, are allowed to influence the predicted warp positions. In previous applications, for which the strip process has been used, studies showed a patch width of ten lines was sufficient for most applications. This provided a correlation coefficient based on a combination of data smoothed over approximately 7.5 lines and independent spatial error signals approximately every 20 lines. To achieve satisfactory results, it was only necessary to control the rate at which these error signals were allowed to update matching positions in the dependent image. For this application, however, it was discovered that the frequency of independence in the error signals had to be more than 20 lines. In addition, it was found that a much larger percentage of the error signal had to be incorporated during the warp update computations to maintain proper registration in areas of rapidly changing terrain. To obtain the results indicated later in this report an effective patch size having a length of 20 pixels and a width defined by a smoothing factor with half-decay time of five lines, and an error signal smoothing factor of six lines, were used. This combination apparently provided the needed flexibility in adjusting position changes due to terrain encountered.

2.2.3 Method of Interlocking Strip Information to Guide Strips in Hard-to-Match Areas

Throughout this discussion, we will refer to the task of guiding strips in hard-to-match areas as harnessing. In previous applications of the strip process, the technique used for this task was as illustrated in Figure 2-5.

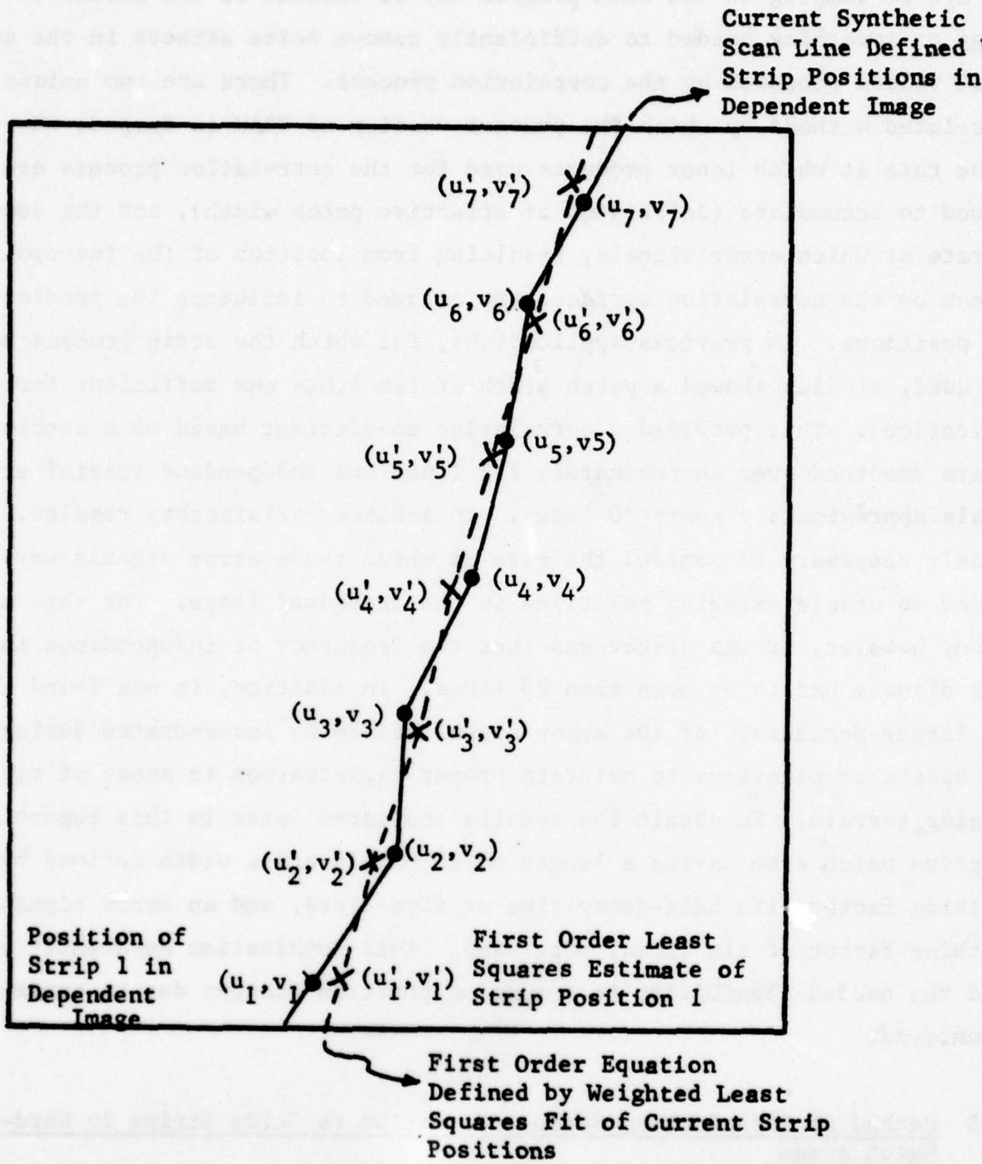


Figure 2-5. Harness Technique For Side Looking Radar

D2379

As the figure suggests, current strip positions are weighted by some measure of their accuracy and a first to fourth order equation (user-defined) is derived using a least squares technique. Current strip positions are compared with least squares estimates and any strips exceeding some circular tolerance limit are corrected by averaging the current position and least squares estimate. Since the harness equation can be first to fourth order in complexity, this technique has proven acceptable in modeling all types of distortion encountered to date.

In the case of stereo images, however, this technique proved unacceptable. The problem becomes one of successfully approximating a frontal curve defined by current strip positions. The order of the equation necessary for this task is a function of the terrain which is currently being matched. As the length of the scan line being matched increases, and the terrain within that scan line varies, one notices that the complexity of the equation necessary to model that scan line becomes increasingly great.

Essentially, what we implemented in order to overcome this problem was a number of piecewise linear approximations to the current frontal curve as illustrated in Figure 2-6. It was found that the information from six strips provided the needed stability for the least squares estimate to control strips within any particular segment of the frontal curve within some circular tolerance, typically three to five pixels. It was our conclusion, however, that this would rarely be the case. In addition, the piecewise linear approximation has the advantage of allowing much greater scan line lengths to be processed without the frontal curve modeling problem inherent in the greater length.

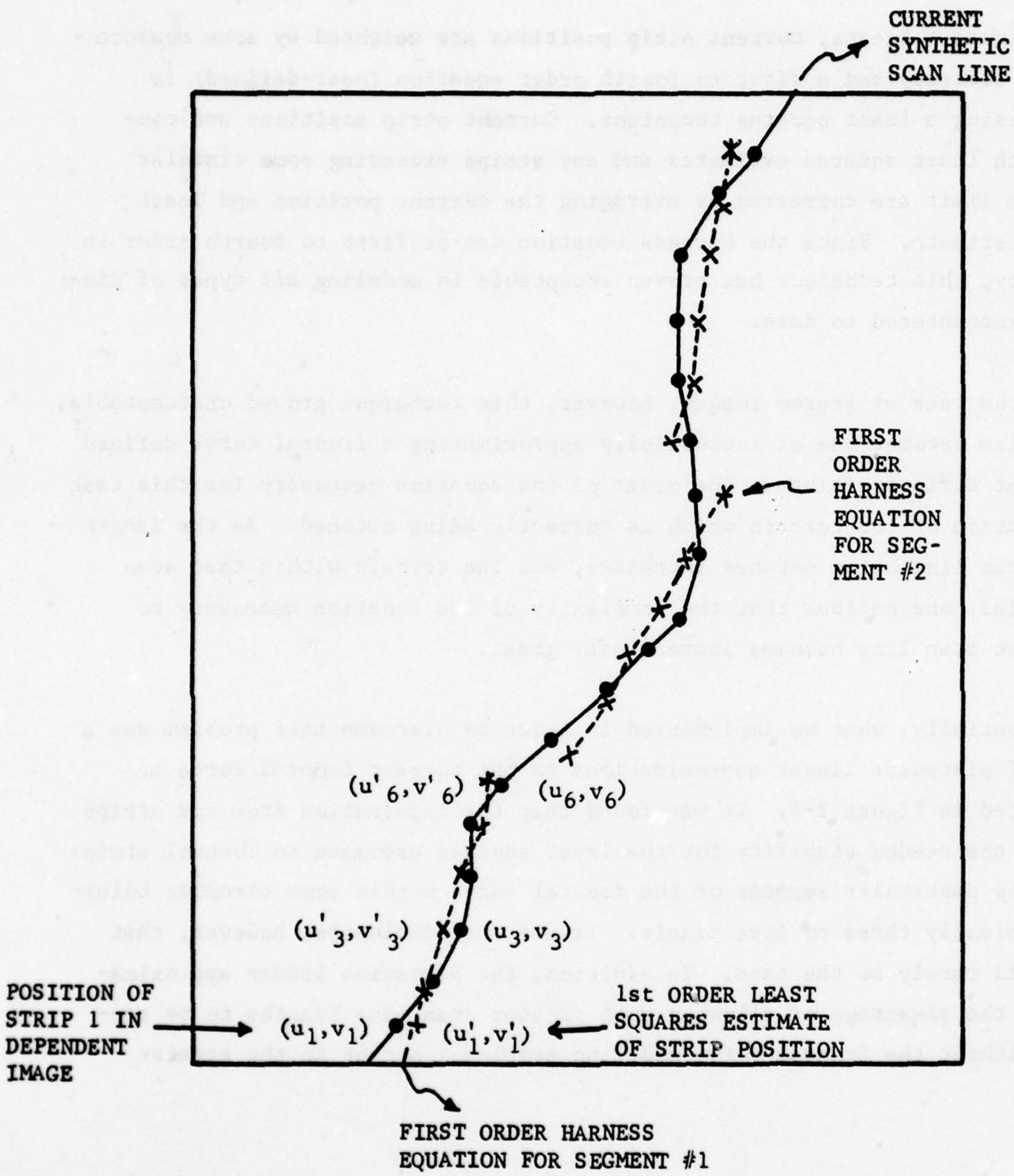


Figure 2-6. Harness Technique Used For Matching Stereo Images

D2377

3.0 MODIFICATION OF BLOCK APPROACH

Under a previous contract, a block processing software system was designed and delivered to USAETL for use with the DIMES System [1]. This block system was research-oriented and capable of being modified in a variety of ways for different applications.

This system has been changed considerably for the present image matching effort, but the underlying philosophy of block processing or area correlation has remained constant. Under this concept, a correlation area of certain dimensions is defined on one image of the stereo pair and a correlation search for the matching area on the other image is initiated. This section describes the particular algorithms that have been substituted or added under the current work effort.

3.1 DATA MANAGEMENT

Previously the block processor was equipped with a data manager that would allow correlation to randomly take place anywhere in the imagery. The cost of this flexibility was a reduction in speed.

Therefore, in the present effort the assumption was made that the imagery is to be traversed in an orderly manner from one end to the other; that is, from scan line 1 to scan line N. This greatly simplifies data management which consists of the maintenance of two buffer windows, one on each image of the stereo pair, that move across the image with the correlation process.

To illustrate this data management, reference is made to Figure 3-1. The buffer window on the independent image, hereafter to be called Image A, is wide enough to include the number of scan lines required for a predetermined patch width. The buffer window on the dependent image, Image B, contains as many scan lines as are required for the patch size search area and the estimated warp between Image B and Image A.

New scan lines are read into the buffers only when needed as the correlation patches progress across the image. Image matching occurs first for all patches in Image A whose centers lie on the same scan lines before the patches move on to subsequent scan lines and before new data is added to the buffers.

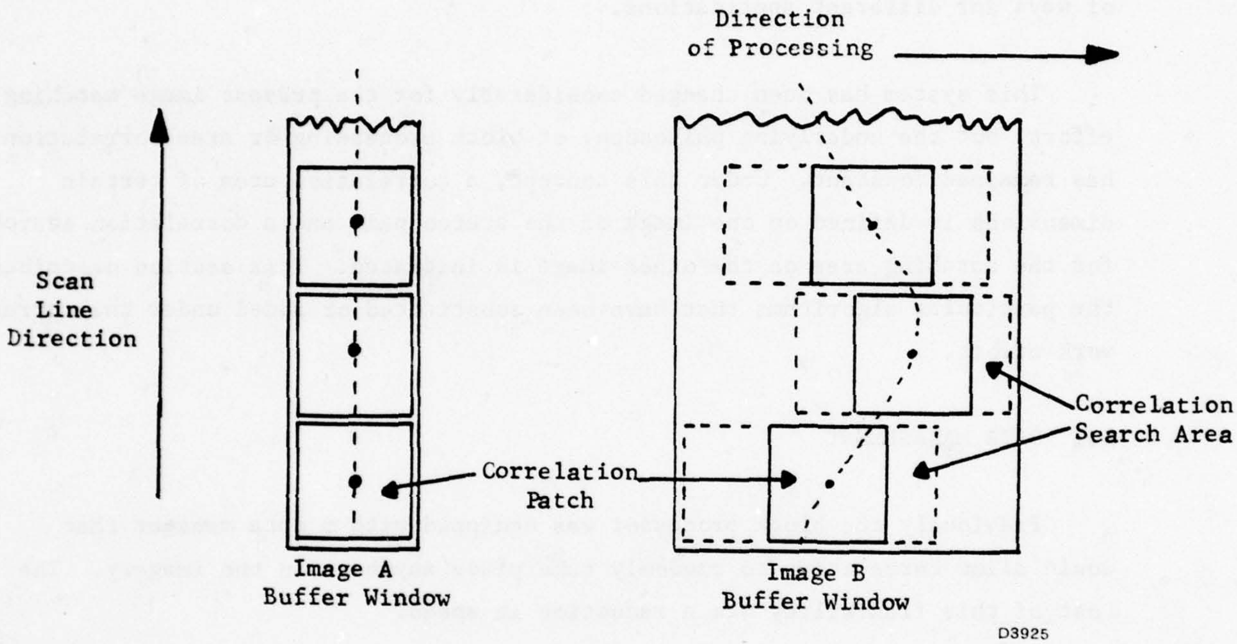
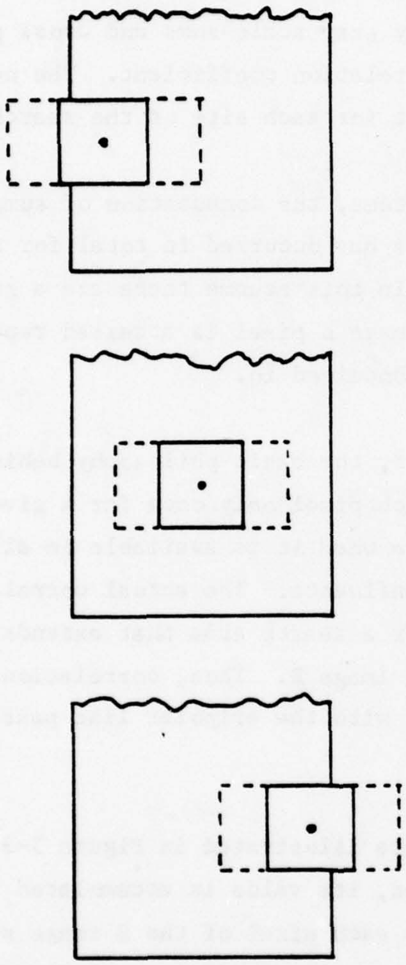


Figure 3-1 Data Management Scheme

Regarding the Image B buffer window, there are three distinct cases of buffer management that can occur for every match point determination. Figure 3-2 depicts these cases.

The optimum situation exists when the buffer window is wide enough to prevent case 1.



Case 1 - Patch and search area behind current buffer window placement-- either enlarge buffer or back up the data

Case 2 - Patch and search area completely enclosed in buffer -- no buffer management necessary

Case 3 - Patch and search area ahead of current buffer window placement -- move window ahead by reading in more scan lines

D3926

Figure 3-2. Three Cases of Image B Buffer Management

3.2 CORRELATION ALGORITHM

In the determination of a match point, it is necessary to place the correlation patch center at each site of the Image B search area and accumulate over the patch area the necessary gray scale sums and cross products needed for the computation of the correlation coefficient. The net result is a value of the correlation coefficient for each site of the search area.

Previously in block matching systems, the computation of sums, cross products, and correlation coefficients has occurred in total for each distinct patch placement in the search area. In this scheme there are a great number of redundant computations, and each Image B pixel is accessed repeatedly, once for each patch placement that it is contained in.

Under the current matching effort, the basic philosophy behind the correlation algorithm is to access each pixel only once for a given search and to accumulate its gray-scale value when it is available in all the sums and cross products for which it has influence. The actual correlation algorithm directs patch placement over a search area that extends in only one dimension around a predicted point on Image B. Thus, correlation values are generated on a line segment coicident with the epipolar line passing through the predicted point.

The mechanics of the algorithm are illustrated in Figure 3-3. As each pixel of the A image patch is accessed, its value is accumulated in the patch sum and sum of squares. Likewise, as each pixel of the B image search area is accessed, its value is accumulated in its respective column sum and column sum of squares. Also, cross products are generated for each patch placement along the Image B search segment. When the accumulation of sums and cross products is complete, the variance of the Image A patch is computed and the array of column sums of squares are traversed to generate an Image B variance and a covariance for each site of the search segment. The array traversal involves the addition of the next column sum and the subtraction of the last column sum from a running total to simulate the movement of the patch along

the search segment.

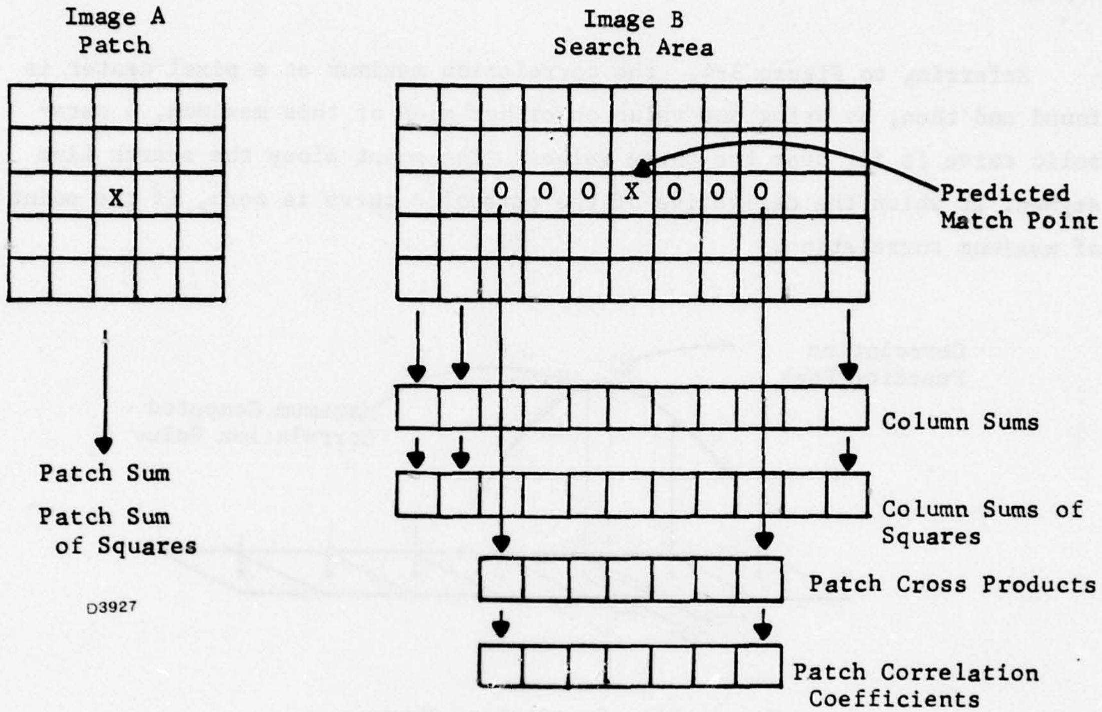


Figure 3-3. Correlation Algorithm Detail

The final result of the algorithm is an array of correlation coefficients of the form

$$r = \frac{\text{cov}(A,B)}{\sqrt{\text{var}(A)\ \text{var}(B)}} ,$$

one value for each placement of the patch along the search segment.

3.2.1 Correlation Maximum Determination

It has been shown in previous studies that it is not sufficiently accurate for cartographic purposes to merely find the maximum value of the correlation coefficient along the search segment at a pixel center.

Rather it is necessary to interpolate the correlation maximum to a fraction of a pixel.

Referring to Figure 3-4, the correlation maximum at a pixel center is found and then, by using one value on either side of this maximum, a parabolic curve is fit over the three values. The point along the search line segment at which the derivative of the parabolic curve is zero, is the point of maximum correlation.

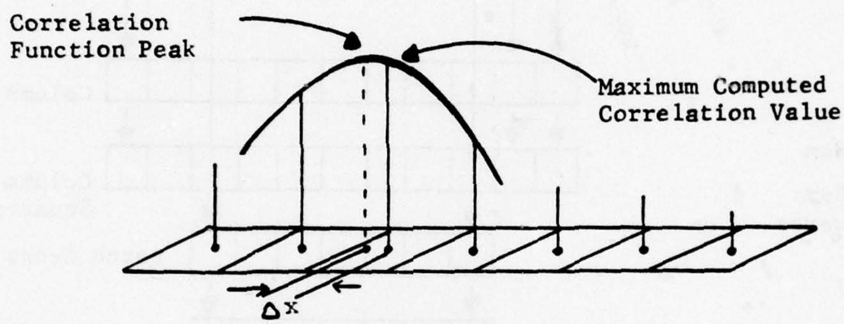


Figure 3-4. Correlation Maximum

A problem exists when the maximum correlation value at a pixel center occurs at either extremity of the search segment. In this case, a parabolic fit cannot be performed, and the algorithm is designed to flag this situation and establish the pixel center as the point of maximum correlation. The underlying idea here is that this match point is suspect and a candidate for further processing.

3.2.2 Match Point Determination

The result of one loop through the correlation algorithm is a pair of conjugate match points of the form (x,y,u,v) where x,y are the coordinates of the point on Image A and u,v are the coordinates on Image B. These points may be expressed by the algorithm in either digital scan coordinates or photo coordinates.

Since the correlation algorithm performs the correlation search in only one direction, that is, along an epipolar line segment, what is really searched and found by gray-scale correlation is the u coordinate. The v coordinate of the match point is computed directly from epipolar geometry parameters; thus eliminating the need for a correlation search in the v direction.

3.3 PREDICTION SCHEME

Prediction in automatic matching systems involves the accurate acquisition of the next point to be used to center a search area based on previously matched points. In the previous block matching systems such prediction was made using a large number of match points (in the neighborhood of 20 to 30). Minimal stereo geometry was used in the prediction, and because of the large number of points used, the prediction was rather global in nature. This required the use of large correlation search areas to compensate for the local image deviations from the global prediction. However, it was found that large, two-dimensional search areas contributed to the instability of the prediction mechanism in hard-to-correlate areas.

Therefore, it was advisable in the current matching effort to design a prediction scheme that:

- uses as much stereo geometry as possible
- is more locally valid
- is accurate enough to reduce the search area to a minimum
- relies less heavily on the correlation coefficient

Such a predictor was implemented and is described in the following sections.

3.3.1 Epipolar Control

As mentioned before, the correlation search for the v coordinate in the former matching system was replaced by the direct computation of the v coordinate from epipolar geometry. Referring to Figure 3-5, the large dots represent previous matched, conjugate points.

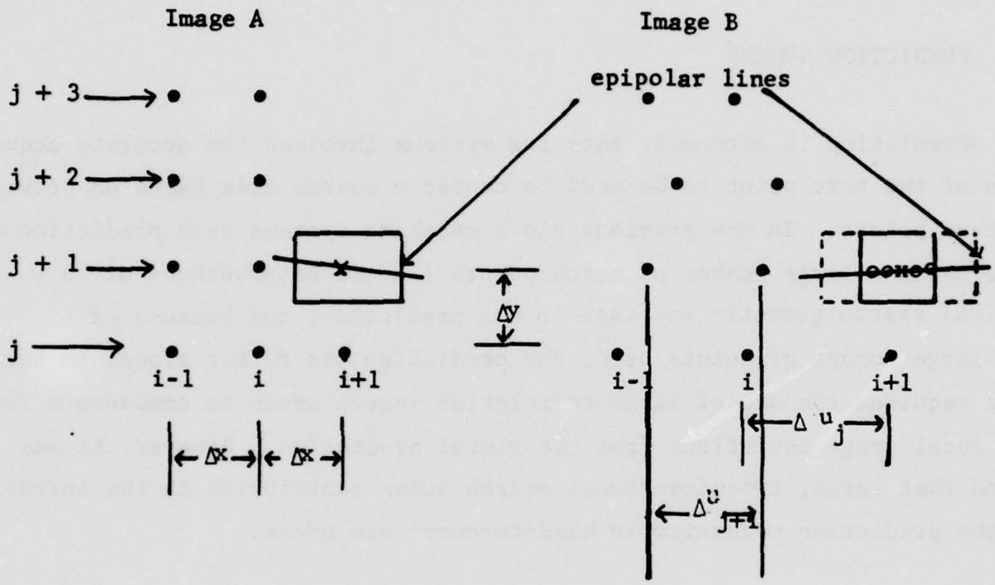


Figure 3-5 Prediction Mechanism

The x's within the patches indicate the points to be matched next. The sequence of matching over the Image A regular grid proceeds from point i,j to i,j+1 to i,j+2, etc. until one line of blocks is complete. Then matching continues with line i+1.

Using the photo coordinates of the center point of the patch i+1, j+1 on the A image and the relative orientation elements of Image B to Image A, the equations of corresponding epipolar lines are determined. Knowledge of where the epipolar line lies on the B Image essentially determines the v coordinate of the match point without need for a correlation search in the

v direction. What is necessary is an accurate prediction of the u coordinate along the epipolar line.

3.3.2 Rate of Change Functions

The scheme used for predicting the u coordinate is based on $\Delta u/\Delta x$, the rate of change of the Image B u coordinate with respect to the Image A x coordinate. This function is kept and updated independently for each path of blocks, j, running along the images in the parallax direction. Δx is the predetermined, regular interval between match points on Image A and Δu's are computed for each path j as new match points are found. In Figure 3-5, for example,

$$\left[\frac{\Delta u}{\Delta x}\right]_j = \frac{u_{i+1} - u_i}{\Delta x} \quad \text{and,}$$

$$\left[\frac{\Delta u}{\Delta x}\right]_{j+1} = \frac{u_i - u_{i-1}}{\Delta x}$$

To obtain a prediction for the u coordinate for match point pair (i+1, j+1), a combination of the neighboring values of $\Delta u/\Delta x$ may be used. For the present image matching effort, the following prediction formula was found most successful for the mountainous areas of the Phoenix imagery:

$$\hat{u}_{i+1,j+1} = u_{i,j+1} + \left(.6 \left[\frac{\Delta u}{\Delta x}\right]_j + .4 \left[\frac{\Delta u}{\Delta x}\right]_{j+1} \right) \Delta x$$

where $\hat{u}_{i+1,j+1}$ is a prediction to be refined by the correlation search. When the correlation algorithm has found the true match point $u_{i+1,j+1}$ along the epipolar line, then $[\Delta u/\Delta x]_{j+1}$ is updated using the new match point for that path and the prediction mechanism moves on to match location (i+1, j+2).

The advantage of such a prediction scheme is that it can be as local or as global as desired, depending upon the number and the placement of

neighboring rate of change functions that are used in a prediction. Moreover, the $\Delta u/\Delta x$ function is proportional to the actual terrain slope as it appears in terms of Image A. The following formulation and its derivations show this relationship:

$$\frac{\Delta u}{\Delta x} = 1 - \frac{\Delta h}{\Delta x}\,\frac{p_i p_{i+1}}{Bf}$$

where $\frac{\Delta h}{\Delta x}$ is the terrain slope with respect to the x coordinate, B is the baseline distance between exposure stations, f is the focal length of the cameras and p_i and p_{i+1} are parallax values over the interval in which the changes are computed. This equation holds only for truly vertical photographs and is set down here to point out that $\Delta u/\Delta x$ varies inversely with the terrain slope and is close to 1.0 in flat terrain. As depicted in Figure 3-6, $\Delta u/\Delta x$ takes on values different from 1.0 depending on whether the terrain is ascending or descending in the process direction.

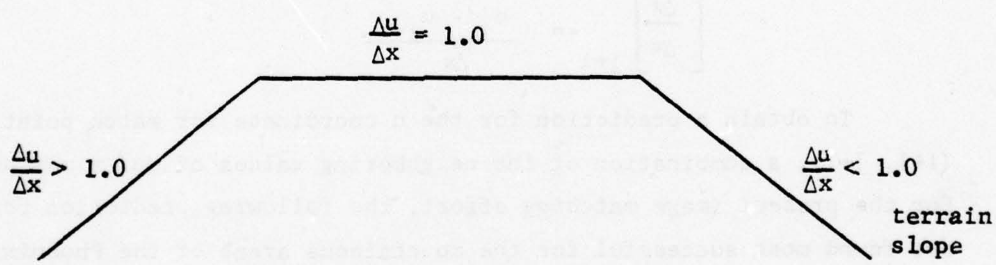


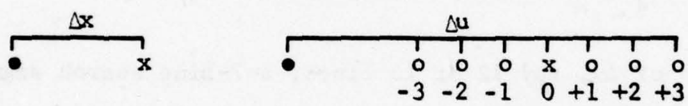
Figure 3-6. Effect of Terrain Slope on $\Delta u/\Delta x$ Function

It has been found that for stereo base/height ratios in the neighborhood of 0.6 and for terrain slopes that do not exceed $\pm$ 45 degrees, the rate of change function is generally within the range:

$$0.5 \leq \frac{\Delta u}{\Delta x} \leq 1.5$$

In terms of automatic matching, the stability of a prediction mechanism using the $\Delta u/\Delta x$ function is very much dependent on both the choice of Δx and the size of the B Image search segment in scan lines. The rationale behind the optimum choice of these parameters is to obtain a prediction within the allowable limits of the $\Delta u/\Delta x$ function so that correlation is only performed in a minimum area. In this way, less emphasis is placed on the correlation coefficient finding a reliable match point over an extended area. Table 3-1 hypothesizes a 7-line search segment centered at a predicted point and summarizes the behaviour of the $\Delta u/\Delta x$ function for given values of Δx and for a correlation maximum occurring at each site of the search segment.

TABLE 3-1. THE EFFECT OF SEARCH SEGMENT SIZE ON THE ΔU/ΔX PREDICTING FUNCTION



Δx (Scan Lines)	Δu (In Lines from Predicted Point)						
	-3	-2	-1	0	+1	+2	+3
	Δu/Δx						
1	-2.0	-1.0	0.0	1.0	2.0	3.0	4.0
2	-.5	0.0	.5	1.0	1.5	2.0	2.5
4	.25	.5	.75	1.0	1.25	1.5	1.75
8	.625	.75	.875	1.0	1.125	1.25	1.375
12	.75	.833	.916	1.0	1.083	1.166	1.25
16	.812	.875	.937	1.0	1.062	1.125	1.187

The table is set up for the case where the center of the search segment is at $\Delta u = \Delta x$, the case of flat terrain starting an inflection upward or downward.

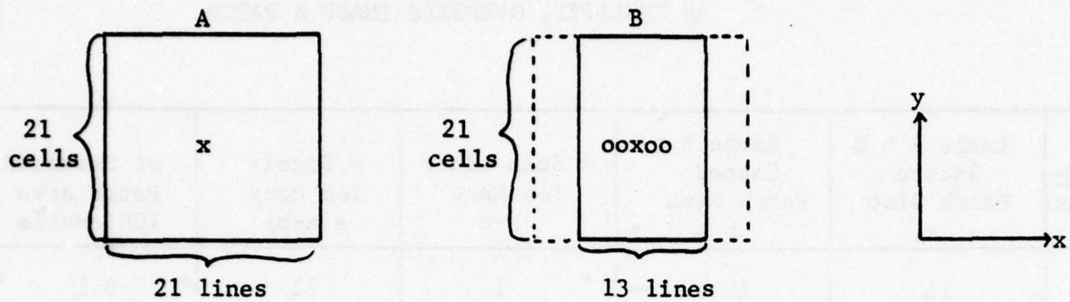
The dashed area of the table indicates the search segment lengths that are critical for maximum stability of prediction for various values of Δx. That is, for Δx values of 2, 4, and 8, the optimum search segment lengths are 3, 5, and 7 respectively. What this means, for example, is that if a 7-line search segment is used with a Δx of 2 lines and if the area being searched is lacking in feature or grossly dissimilar on both images, then the probability is great that the correlation algorithm will find a correlation maximum toward the ends of the search segment. Thus, the $\Delta u/\Delta x$ function value becomes too low or too high, resulting in a biased next prediction. It has been observed in these cases that succeeding correlation maximums along a path are found alternately at one end of the segment and then the other, causing the $\Delta u/\Delta x$ function to oscillate rapidly until good feature lock-on can be achieved or until the prediction mechanism totally degenerates.

For larger values of Δx, say 12 or 16 lines, a 7-line search segment is unresponsive to rapid terrain changes. In these cases, the search segment length must be increased. For the current matching effort, a 7-line search segment with a Δx value of 8 lines has been found to produce the best results. Of course, this is very dependent upon the scale of the imagery and the type of terrain being matched.

3.3.3 Correlation Patch Shaping

The $\Delta u/\Delta x$ function is not only a valid metric for prediction purposes but also for measuring the amount of local Image B compression or expansion relative to the same area on Image A. In previous block matching systems, a correlation search was performed using patches of equal size on the A and B Images. It was found, however, that this procedure is valid only in flat terrain, that is, where $\Delta h/\Delta x$ is close to zero and $\Delta u/\Delta x$ is close to 1.0. For all other cases, the B Image patch must be compressed or expanded primarily in the major parallax direction. Referring to Figure 3-7 as an example, if the size of the Image A patch is chosen as 21 x 21 cells and the current $\Delta u/\Delta x$ function is .6, then the corresponding patch on the B Image is

21 cells by 13 lines. The Image B patch and search area height remains at 21 cells because the image y parallax is negligible in terms of whole pixel intervals.



$$\frac{\Delta u}{\Delta x} = .6$$

Figure 3-7. Patch Shaping

As a further sophistication of patch shaping, the B Image patch sides (the 21-cell dimension of the figure) may be slanted to account for Image B warp due to terrain slope in the y direction.

The correlation coefficient for these patches must be computed using 21 x 21 pixel samples. Therefore, the Image B patch area must be resampled to derive 21 x 21 gray-scale values from the 21 x 13 pixels that are contained in the patch area. In the current implementation of the block matching system, the entire search area is shaped according to the $\Delta u/\Delta x$ function and a synthetic search area is generated which contains the required number of samples for the correlation algorithm.

The need for patch shaping in other than flat terrain is corroborated in Table 3-2. The table hypothesizes that the correlation patch size is the same on both the A and B Images, and computes the percentage of unwanted gray-scale samples that effect the correlation coefficient over terrain sloping upward in the process direction. B Image patch shaping eliminates these unwanted samples.

TABLE 3-2. THE AMOUNT OF STATISTICAL INFLUENCE OF AN UNSHAPED, OVERSIZE IMAGE B PATCH

$\frac{\Delta u}{\Delta x}$	Image A & B Square Patch Size a	Image B Shaped Patch Size b	# Scan Lines Too Many a-b	# Pixels Too Many a(a-b)	% of Unwanted Patch Area 100(a-b)/a
.9	11	10	1	11	9.1
	15	14	1	15	6.7
	21	19	2	42	9.5
	25	23	2	50	8.0
	31	28	3	93	9.7
.8	11	9	2	22	18.2
	15	12	3	45	20.0
	21	17	4	84	19.0
	25	20	5	125	20.0
	31	25	6	186	19.4
.7	11	8	3	33	27.3
	15	11	4	60	26.7
	21	15	6	126	28.6
	25	18	7	175	28.0
	31	22	9	279	29.0
.6	11	7	4	44	36.4
	15	9	6	90	40.0
	21	13	8	168	38.1
	25	15	10	250	40.0
	31	19	12	372	38.7

A secondary conclusion that can be drawn from the table is that the need for patch shaping is independent of patch size. However, this is valid only when the $\Delta u/\Delta x$ function remains constant over the patch area. Also, the larger the patch, the more non-linear the sides of the patch in the y direction become.

3.4 DISSIMILAR IMAGE AREAS

As described earlier, correlation subregions move across the images in the major parallax direction in paths. Each path is basically independent of the others, being controlled by its own $\Delta u/\Delta x$ function. But because the paths are independent, it is not uncommon in featureless areas or areas that are dissimilar in gray-scale on the two images, for blocks in a given path to wander, that is, fall excessively behind or move ahead of blocks in adjacent paths. The wandering usually occurs when the lack of correlation lock-on causes the $\Delta u/\Delta x$ function for that path to become unstable.

Therefore, a mechanism has been provided in the current block matching system to detect wandering blocks and to correct their position, based on blocks in adjacent paths. This harnessing technique is shown in Figure 3-8.

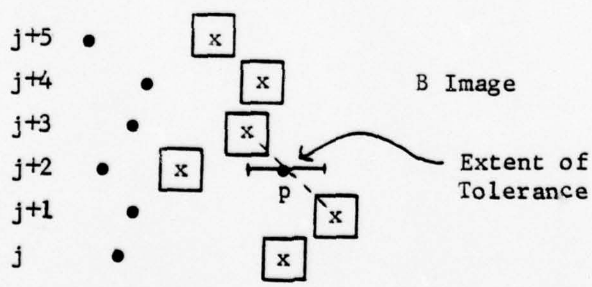


Figure 3-8. Wandering Block

When the processing of a complete line of blocks (blocks which have the same digital x coordinate in Image A) is complete, a check is made to see that all blocks lie within a predetermined tolerance distance from the

average position of the blocks in adjacent paths. If not, the position of the wandering block and the $\Delta u/\Delta x$ function for that path are corrected accordingly. Referring to the figure, the block on path j+2 has been detected as wandering. Its position is corrected to point P, the average position of paths j+1 and j+3.

A special case exists for the end paths, paths j and j+5 of the figure, where complete harnessing is not possible. A problem also exists when blocks in more than one path begin to wander. The probability of this occurance on good imagery is rather low, but when it does occur, the described mechanism can correct the wandering blocks iteratively over a longer stretch of imagery. When all paths become lost, there is not much that can be done.

4.0 PROCESSING RESULTS

A digital stereo pair of images was received from USAETL with all the appropriate interior orientation transformations and exterior orientation parameters. The digital image data represents a four-square-inch scene from the 1:48000 Phoenix-South Mountain model. Digital encoding was performed at ETL with a 35 micrometer spot size and a 24 micrometer sampling interval. A pixel, therefore, nominally represents four feet on the ground.

Supplied with the imagery was also a file of 25,250 match points generated by an ETL block process. These match points covered an image area of approximately .25 inches by .5 inches at original image scale and were used to initialize both the CDC strip process and block process.

The modified strip process and block process as described in the sections above were applied to the same image area, and files of match points were generated. For the strip processing, the strip centers were located 10 pixels apart and the strip width was chosen as 20 pixels. The effective recursive correlation area along the strips was chosen as six scan lines. Even though the strip process provides line-by-line tracking of an image, match points were output every eight scan lines, thus providing a match point lattice whose interval is ten pixels in the y direction by eight lines in the x direction.

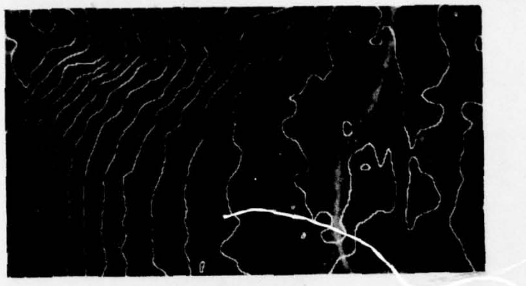
In both processes, Frame 98 of the stereo pair was chosen as Image A, the independent image, and Frame 99 as Image B or the dependent image. For the block process, a correlation patch size of 21 x 21 pixels on Image A was chosen. The distance between blocks or block paths on Image A was ten pixels and Δx or the block jump-interval was eight lines. A seven-line search segment was used on Image B, including three scan lines on either side of a predicted point.

As a result, three match point files were available for analysis; the ETL file, the block file, and the strip file. These files were processed using photogrammetric intersection to obtain for each point an orthometric terrain

elevation in feet above the earth. These elevations were given X,Y coordinates corresponding to the digital raster of Image A. Therefore, the resultant terrain data is not truly orthographic, but rather as unrectified as Image A.

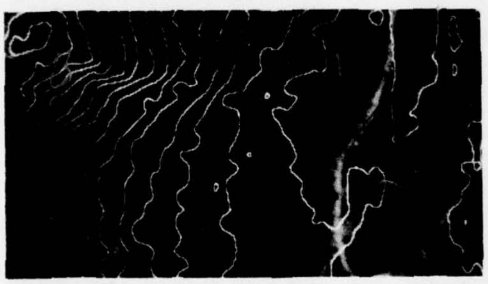
These files of terrain data were fit with local, smooth bicubic polynomials to produce the contour lines shown in Figure 4-1(a,b,c). The contour interval is 20 feet and the background is the Image A section. This section of image has been enlarged eight times, the original scale area being approximately .25 inches by .5 inches. To point out the differences between approaches, contour difference images were generated and appear in Figure 4-1 (d,e,f). These difference images were produced by subtracting corresponding contour region images, where the 20-foot contour regions are alternately colored black and white. The neutral gray color indicates similarity between contour regions or zero in the subtraction. The white and black contour differences change polarity on every contour region. Therefore, visual reference must be made to Figure 4-1(a,b,c) to determine which difference in Figure 4-1 (d,e,f) came from which image. It must be kept in mind that because the terrain elevations have been quantized into discrete 20-foot intervals to produce the difference images, the actual magnitudes of the differences observed can range between 0 and 20 feet. As a quantitative measure, the RMS magnitude of the actual orthometric elevation differences is in the neighborhood of eight feet for all three cases.

The processing of the Phoenix model using the strip approach and block approach was extended beyond the ETL match point data to include about 12,000 match points across the entire length of the digital sample. The results of this processing in contour line form along with the contour differences appear in Figure 4-2. These images are enlarged four times. The actual image area is approximately .5 by two inches at the original scale of the photography.



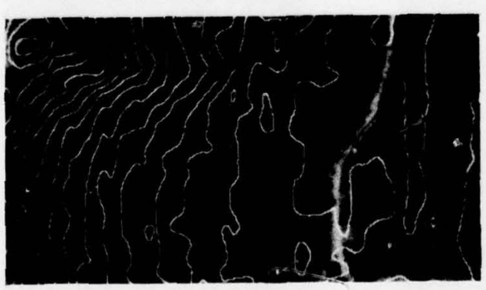
(a) ETL Approach

(d) ETL vs. Block Approach



(b) Block Approach

(e) ETL vs. Strip Approach

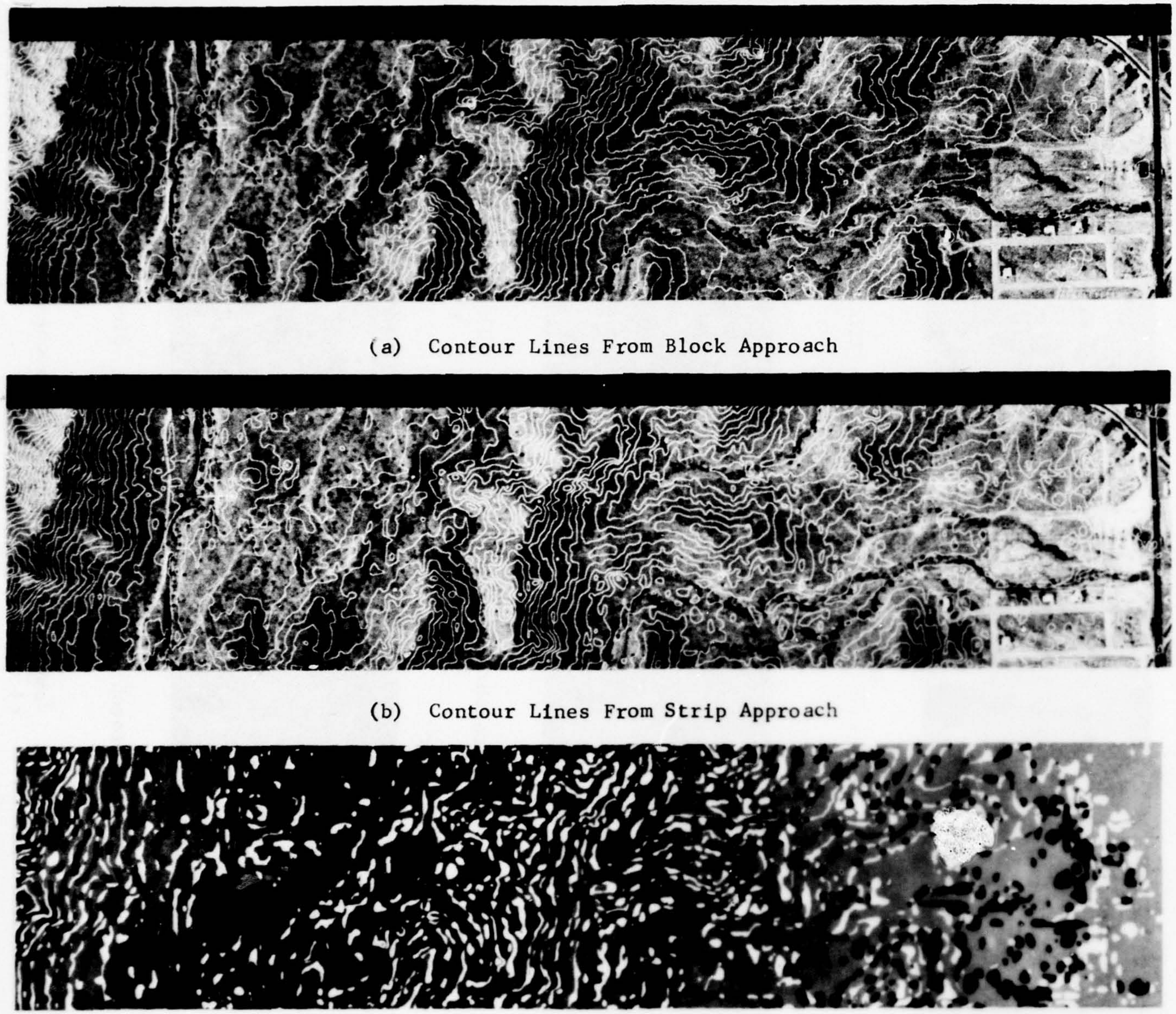


(c) Strip Approach

(f) Block vs. Strip Approach

Figure 4-1. Comparison of Matching Approaches in Terms of Contoured Terrain Data

a, b, c - Contour Images, 20 feet interval
d, e, f - Contour Difference Images



(a) Contour Lines From Block Approach

(b) Contour Lines From Strip Approach

(c) Contour Differences - Block Approach vs. Strip Approach

Figure 4-2. Comparison of Matching Approaches Over a Larger Area

A by-product of the strip approach is the warping of the B Image to register with the A Image and also the generation of a tonal difference image for change detection purposes. This difference image and the corresponding unrectified section of Image B as reference are shown in Figure 4-3.

The tonal difference image is a qualitative measure of the goodness of match for the strip approach. As can be seen in the image, the only real gray scale differences after registration are the reseau marks and the dissimilar appearance of the tall buildings in the right of the scene. Referring back to Figure 4-2, it can also be seen that some of the buildings were contoured as part of the terrain by the strip approach. But despite this goodness of registration, the actual RMS differences in elevation between the strip approach and the block approach for the 12,000 match points was again about 8 feet.

As a further demonstration of the performance of the two approaches, the raw terrain data, without fitting or smoothing, that was produced by intersection from the match points was plotted in a three-dimensional mode on a Calcomp plotter. These plots appear in Figures 4-4 and 4-5. Each plotted line in these figures represents one profile in the y direction. As was mentioned before, the interval between profiles is 8 scan lines. The vertical dimension in these plots has been greatly exaggerated to emphasize small-scale differences between the approaches. It can be seen that the strip process produces a noisier data set.

Regarding processing speed, the general-purpose implementation of the block approach as described above produces 14 match points per central processor second. The strip approach produces 27 match points per second. This figure for the strip approach is based on a match point every 8 scan lines. But because the strip process performs correlation on a scan line-by-scan line basis, the capability exists for generating a match point for each strip on every scan line at no extra time. Therefore, the process can produce up to 216 match points per second. Both the block process and strip process are implemented in FORTRAN running on the CDC 6600 computer system.

(a) Strip Process Tonal Difference Image

(b) Original Section of Frame 99

Figure 4-3. Strip Process Registration Evaluation

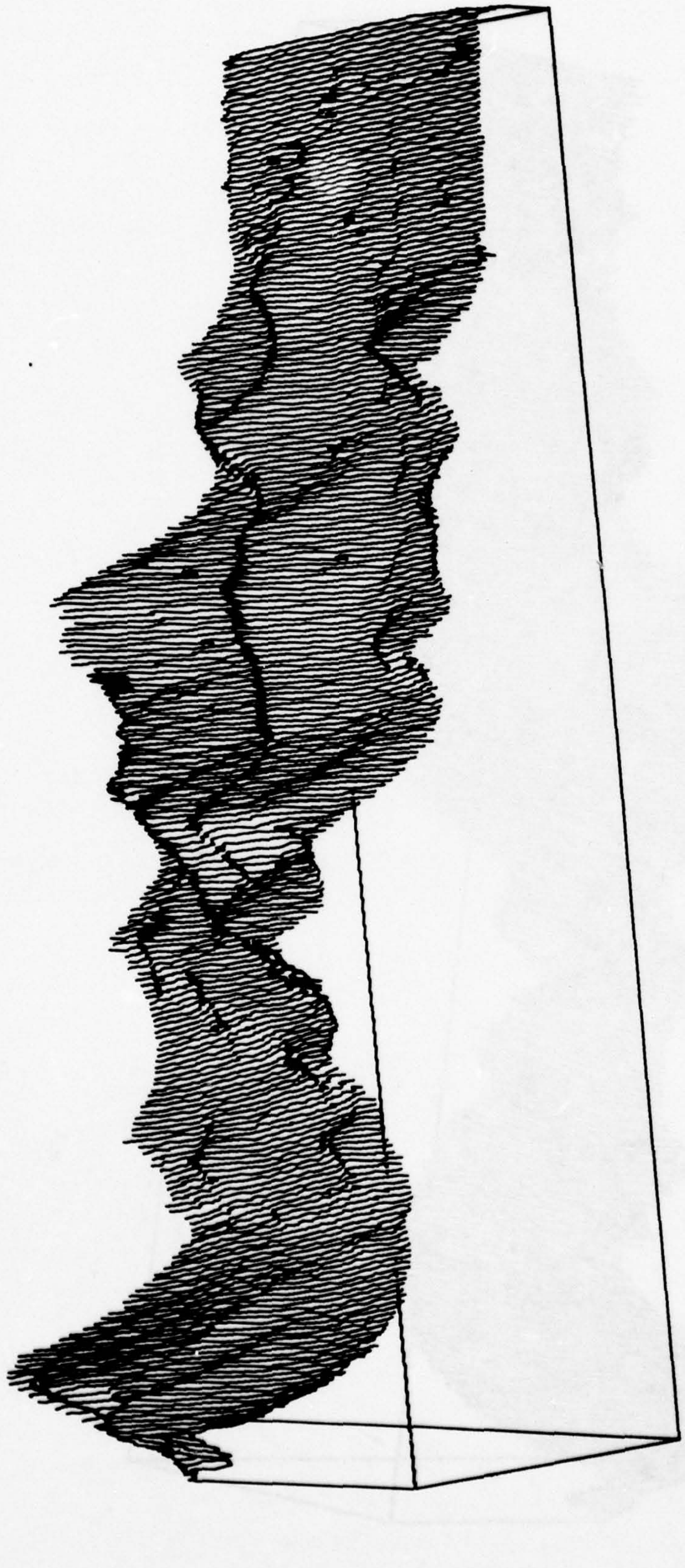


Figure 4-4. 3-D Plot of Block Process Terrain Data

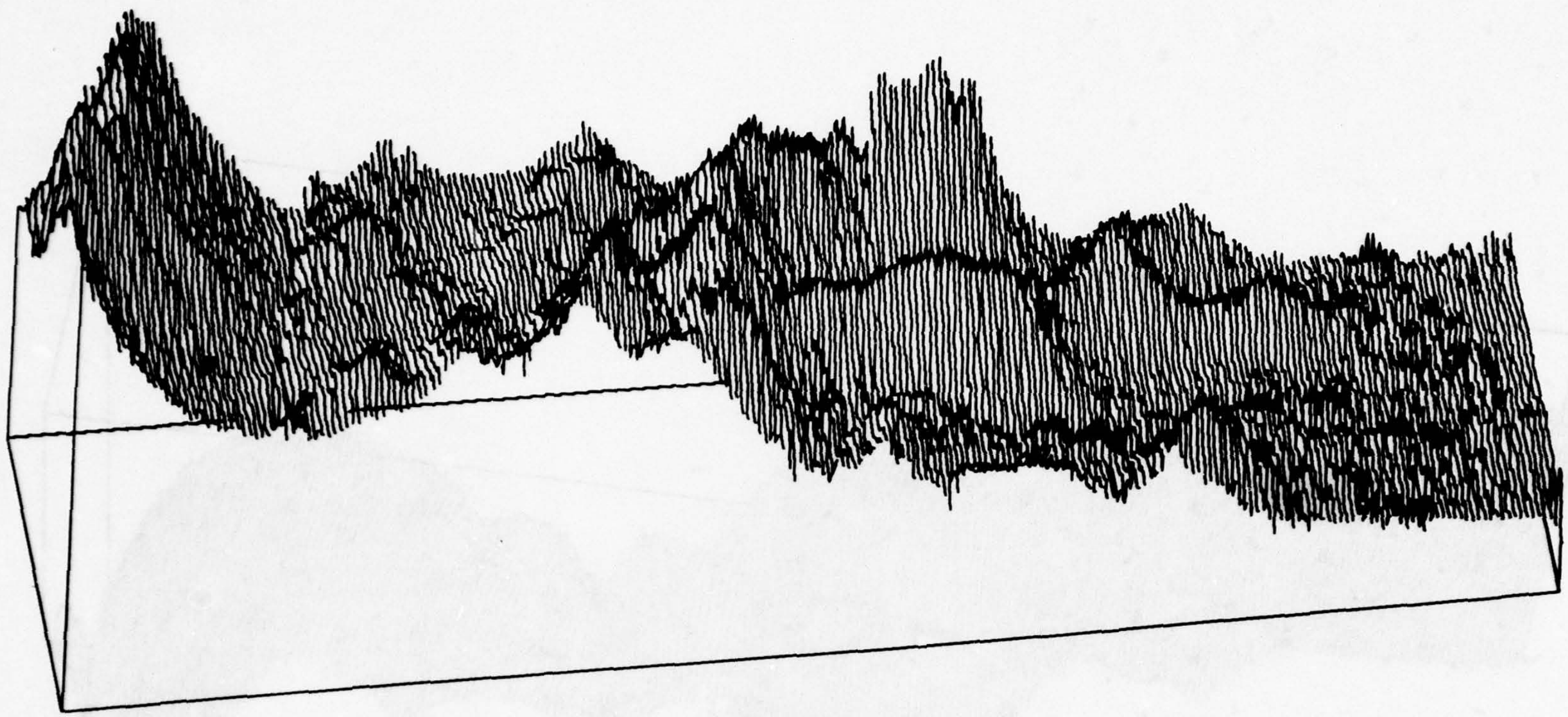


Figure 4-5. 3-D Plot of Strip Process Terrain Data

5.0 CONCLUSIONS

It can be seen from the contour images presented above that both the strip approach and the block approach in their present implementations are following the general terrain trends of the imagery. It is conjectured that if the terrain data from the two approaches were evaluated against known ground or model control points using standard photogrammetric techniques, the RMS deviations at the control points would be very small. This is due to the fact that control points are generally associated with well-defined features--features which can be matched very well by an automatic process. It is the featureless, between-control-point areas that pose problems both for automatic matching and terrain data evaluation.

The strip approach is a very fast production-oriented process. But the digital terrain data it generates is rather noisy. This seems to be an attribute of the line-by-line, error-correcting nature of the process. By placing constraints on the process smoother terrain data results, but the process becomes less responsive to terrain fluctuations. That is, the process tends to overshoot mountain peaks and to dig into valleys.

The block approach, on the other hand, generates less noisy terrain data due to its deterministic predict-ahead mechanism. The values of the correlation coefficient are generally higher in the block approach than in the strip approach and the match points generated in feature-rich image areas seem to be more accurate in the block approach. However, in featureless and dissimilar image areas, the block process has less system inertia to carry it through the difficult regions than the strip process. In addition, the block process is slower.

With all these tradeoffs in mind, the block process is being considered further for implementation in the digital cartographic benchmark.

REFERENCE

1. Bonrud, L. O. et al., "Analysis and Development of Digital Mapping System Software," Final Report to U. S. Army Engineer Topographic Laboratories, ETL-CR-74-5, May 1974.

DIGITAL CARTOGRAPHIC STUDY AND BENCHMARK
SECOND INTERIM TECHNICAL REPORT

Prepared for:

U. S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia

Contract DAAG53-75-C-0195

Prepared by:

D. J. Panton
M. E. Murphy

December 1975

Control Data Corporation
DIGITAL IMAGE SYSTEMS DIVISION
2800 East Old Shakopee Road
Minneapolis, Minnesota 55420

TABLE OF CONTENTS

LIST OF FIGURES

1.0 INTRODUCTION

This technical report is a summary of the work performed during the second phase of an ongoing three-phase effort. The purpose of the effort is to assess the practicality of implementing the automatic process of stereo image matching on a configuration of extremely fast microprogrammable processors.

In general, the three-phase program is designed to provide the researcher with a means for bridging the gap between mathematical algorithm development and an actual hardware benchmark realization. The steps of this generalized program are illustrated in Figure 1-1.

In the present context of a stereo image matching benchmark, both Phase A and Phase B have been completed. The objective of Phase A was to redesign and modify in a general-purpose software environment two existing image matching approaches, the strip approach and the block approach. As a result of Phase A, it was determined that the block approach best suited the requirements of and the flexibility necessary for digital stereo mapping [2].

The goal, then, of Phase B was to break the block process apart into its logical components in such a way that its implementation in microcode would be straightforward. The justification for this lies in the fact that algorithms that have been developed in a higher level programming language and that run on general-purpose sequential machines are generally too slow to to be practical in a large data volume, image processing environment. What is practical is the incorporation of the desired algorithm in a hardwired machine that is capable of the utmost process parallelism and processing speed. However, this method of implementation is rather inflexible as regards algorithm modification and machine versatility. The availability of arrays of microprocessors at first seems to offer the researcher the best of both worlds; the speed of microprocessors can approach that of hardwired machines and flexibility is maintained in that the processors are microprogrammable.

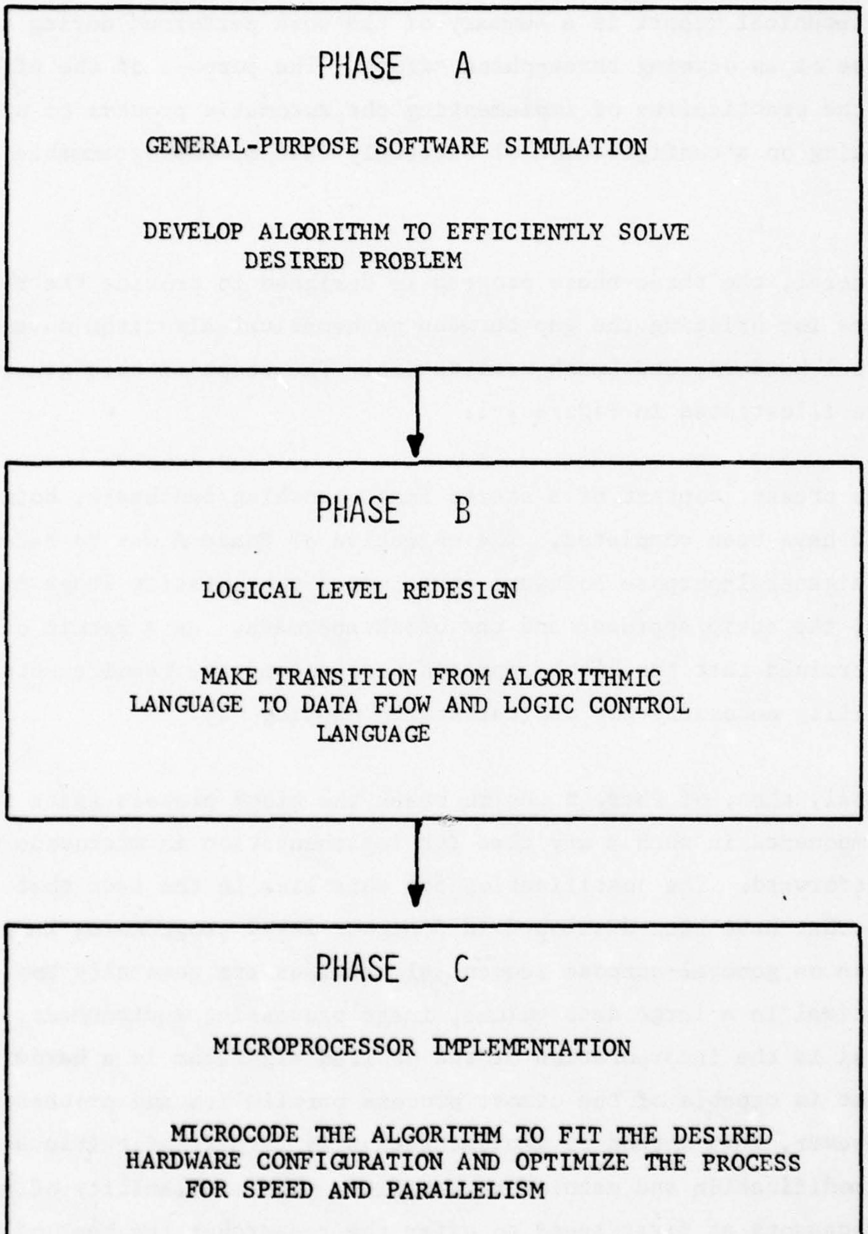


Figure 1-1. Steps of Three-Phase Program

However, microprogramming is performed in a data flow and machine logic control language whereas the algorithm was developed in a mathematical, problem-oriented language. The direct transliteration of the higher level language into microcode does not result in a high performance realization of the algorithm. What is necessary is the reconstruction of the algorithm in terms of its data flow and logical component interconnections.

The following sections of the report describe this reconstruction of the block processing approach to stereo image matching. These results will be used in Phase C of the current effort to actually benchmark the process on an array of microprocessors.

2.0 ALGORITHM REFINEMENTS

In an effort to firm up the block processing algorithm for logic level reconstruction, two refinements were made that were not included in the technical report for Phase A. One involved the addition of a reliability measurement capability, and the other involved the further sophistication of the correlation patch shaping scheme.

2.1 MATCH POINT RELIABILITY DETERMINATION

It is desirable for analysis purposes to obtain for each match point generated by the block process a measure of the point's reliability in terms of internal matching parameters at the time the point is being generated. Therefore, a reliability factor was devised which characterizes the reliability of a match point with respect to the following criteria:

- value of the correlation coefficient
- gray-scale standard deviation of the A image patch
- distance of the correlation maximum from the predicted location
- range limites of the prediction function
- slope of the correlation function

The form of this reliability factor is illustrated in Figure 2-1.

As is shown, the reliability factor is an N digit number, one digit that is either 1 or 0 for each reliability criterion. In implementation, each criterion value for a match point is tested against a preset cutoff value. If the test fails, a 1 is placed in the factor for that criterion. Therefore, a match point that is reliable with respect to all the criteria has a reliability factor of zero. At the end of the matching run a reliability summary is printed which includes both the percentage of totally reliable points and the percentage of unreliable points with respect to each criterion.

The reliability factor as described performs a number of useful analysis functions both in the continued modification of the algorithm and in the final evaluation of match points. If the analyst working on the algorithm discovers a new technique that he feels is theoretically effective for a section of the process, he may observe the effect of this technique directly in the reliability summary after he has made the change in the algorithm, all other things being the same in the matching run. As an example, the patch shaping algorithm described in the next section was thought to be more accurate than the algorithm that was currently implemented. The new algorithm was inserted in the block matching code and the result was that the number of totally reliable match points increased from 78% to 96%, verifying that for that particular data the new algorithm was clearly superior.

As an internal aid to the matching process the reliability factor for each match point can provide the information necessary for rematching certain points. This falls under the matching philosophy that an attempt be made to improve unreliable points as they are generated, when the process has at its disposal all that is to be known about the points, rather than allowing the process to roar through the data, leaving unreliable points to be resolved by a postprocess.

In rematching, the individual digits of the reliability factor act as a decision table. For instance, if on a given point the value of the

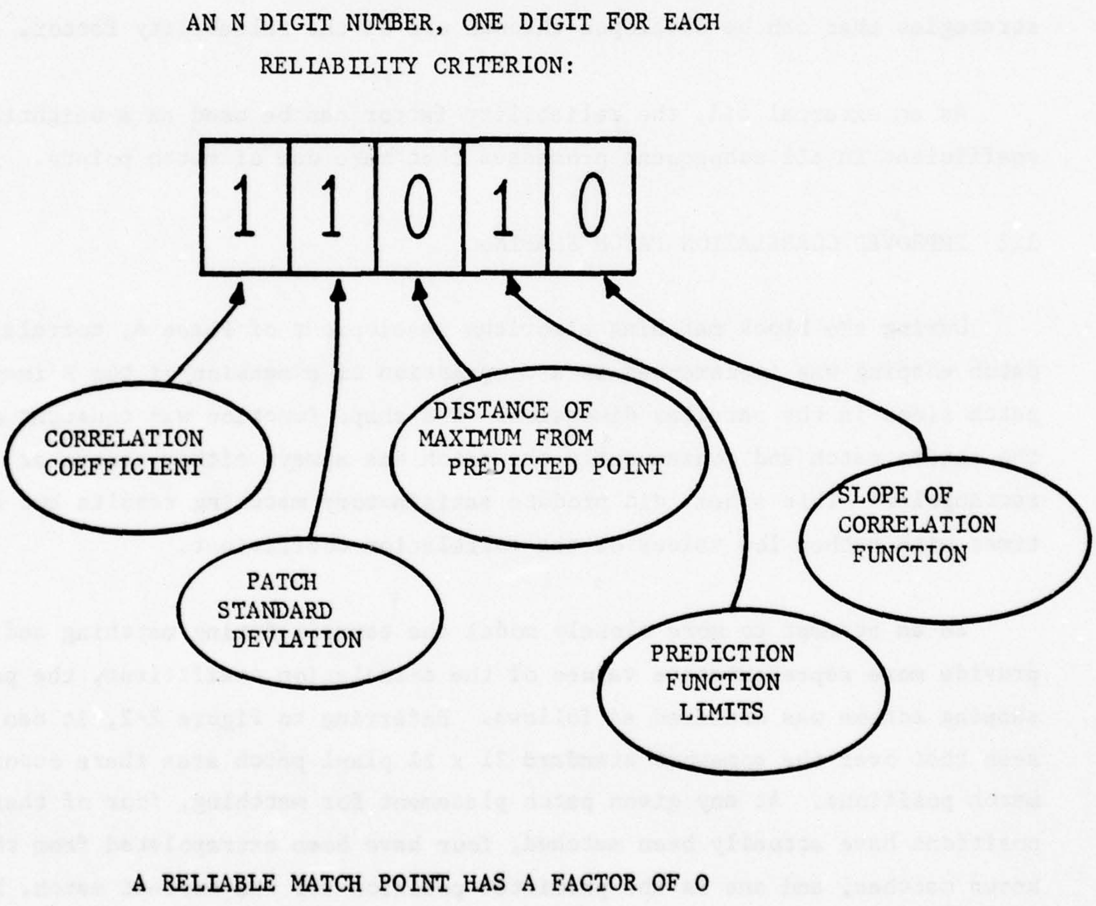


Figure 2-1. Match Point Reliability Factor

correlation coefficient is low and the standard deviation is low, then rematching of the point should occur with a larger patch size. When the rematch is complete, the new reliability factor can be compared with the old to determine if improvement has occurred. In another situation, if the value of the correlation coefficient is low and if the distance of the correlation maximum from the predicted maximum is great or if the prediction function has exceeded its limits, then rematching should occur around a new predicted point and consequently with a new patch shaping. These are but two of the strategies that can be developed through use of the reliability factor.

As an external aid, the reliability factor can be used as a weighting coefficient in all subsequent processes that make use of match points.

2.2 IMPROVED CORRELATION PATCH SHAPING

During the block matching algorithm development of Phase A, correlation patch shaping was implemented as a compression or expansion of the B image patch sides in the parallax dimension. The shape function was constant over the entire patch and consequently the patch was always either square or rectangular. This scheme did produce satisfactory matching results but sometimes with rather low values of the correlation coefficient.

In an attempt to more closely model the terrain during matching and to provide more representative values of the correlation coefficient, the patch shaping scheme was modified as follows. Referring to Figure 2-2, it can be seen that over the somewhat standard 21 x 21 pixel patch area there occur nine match positions. At any given patch placement for matching, four of these positions have actually been matched, four have been extrapolated from the known matches, and one is the predicted position for the current match. These nine positions are evenly spaced on the A image patch, but they serve as inflection points for shaping the B image patch. As is shown in the figure, both the rate of change prediction function $\Delta u/\Delta x$ and the scan line starting position can vary for each of the three rows of three points. These differentials have been taken into account in shaping the B image patch under the

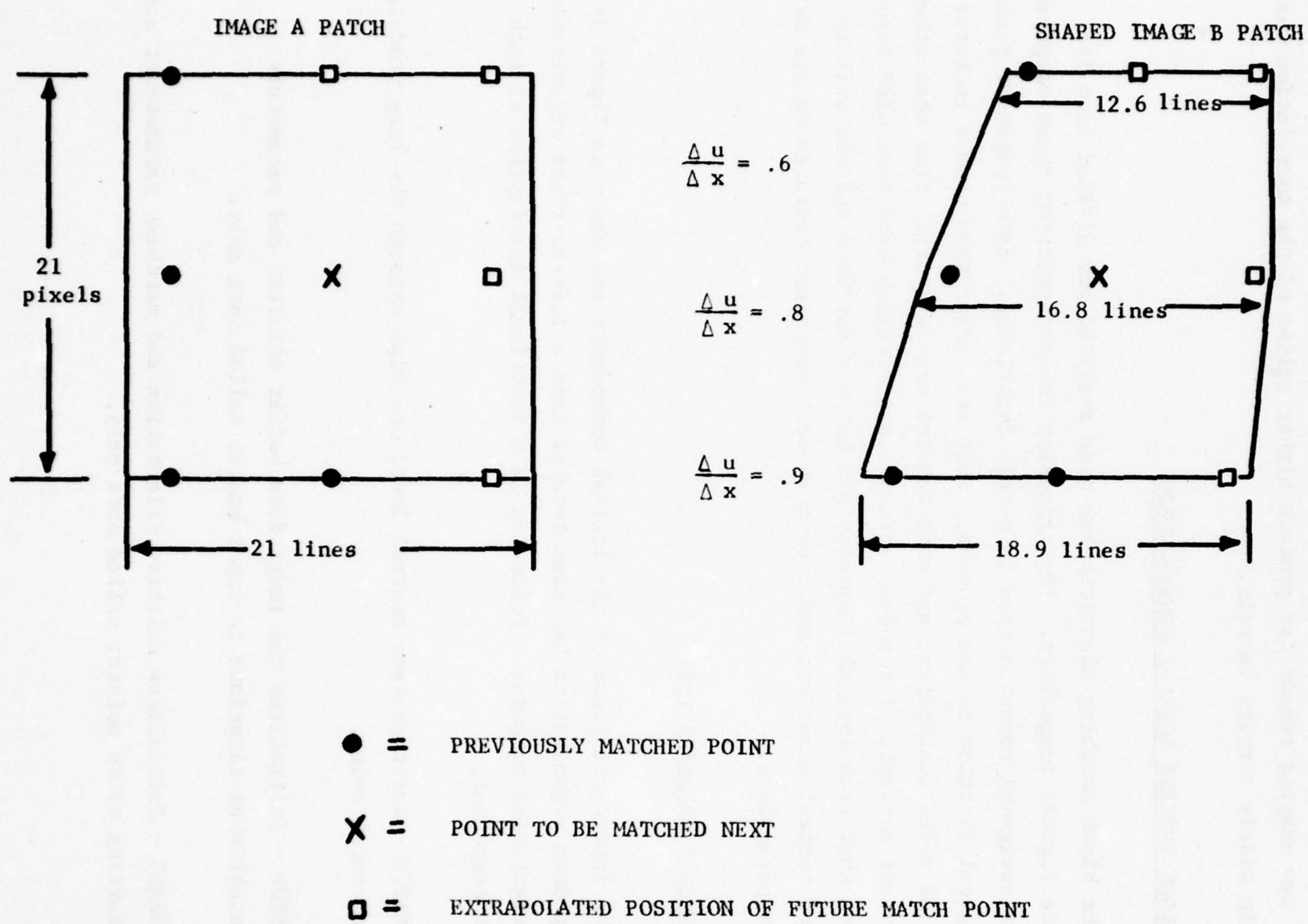


Figure 2-2. Correlation Patch Shaping

new scheme. As a result, the B image patch is more nearly a six-sided figure and has more degrees of freedom in conforming to the terrain. The inclusion of the new shaping scheme did produce higher values of the correlation coefficient in widely varying terrain.

3.0 BLOCK PROCESS LOGICAL CONSTRUCTION

The block matching algorithm has been analyzed and divided into its discrete logical components. The criterion for constructing these components and for assigning names to them is purely functional. Each component performs one logical function in the process. Some are larger than others in terms of number of coded statements, and some require more execution time than others. The current concepts of top-down software organization have been used throughout such that each logical component has but one entrance and one exit in terms of control structure and such that each component operates on one set of data or parameters.

3.1 PROCESS ORGANIZATION

The interconnections of the logical components are shown in Figure 3-1. The top-down organization has been rotated into a left-to-right organization for presentation purposes. Following is a functional description of each logical component.

BMATCH - Basic process control loop; one time through the loop produces one match point.

SETUP - Initializes the image data buffer pointers and parameters; establishes interface to input units; called once only.

PARAMS - Initializes relative orientation and matching parameters; accepts starting match points; called once only.

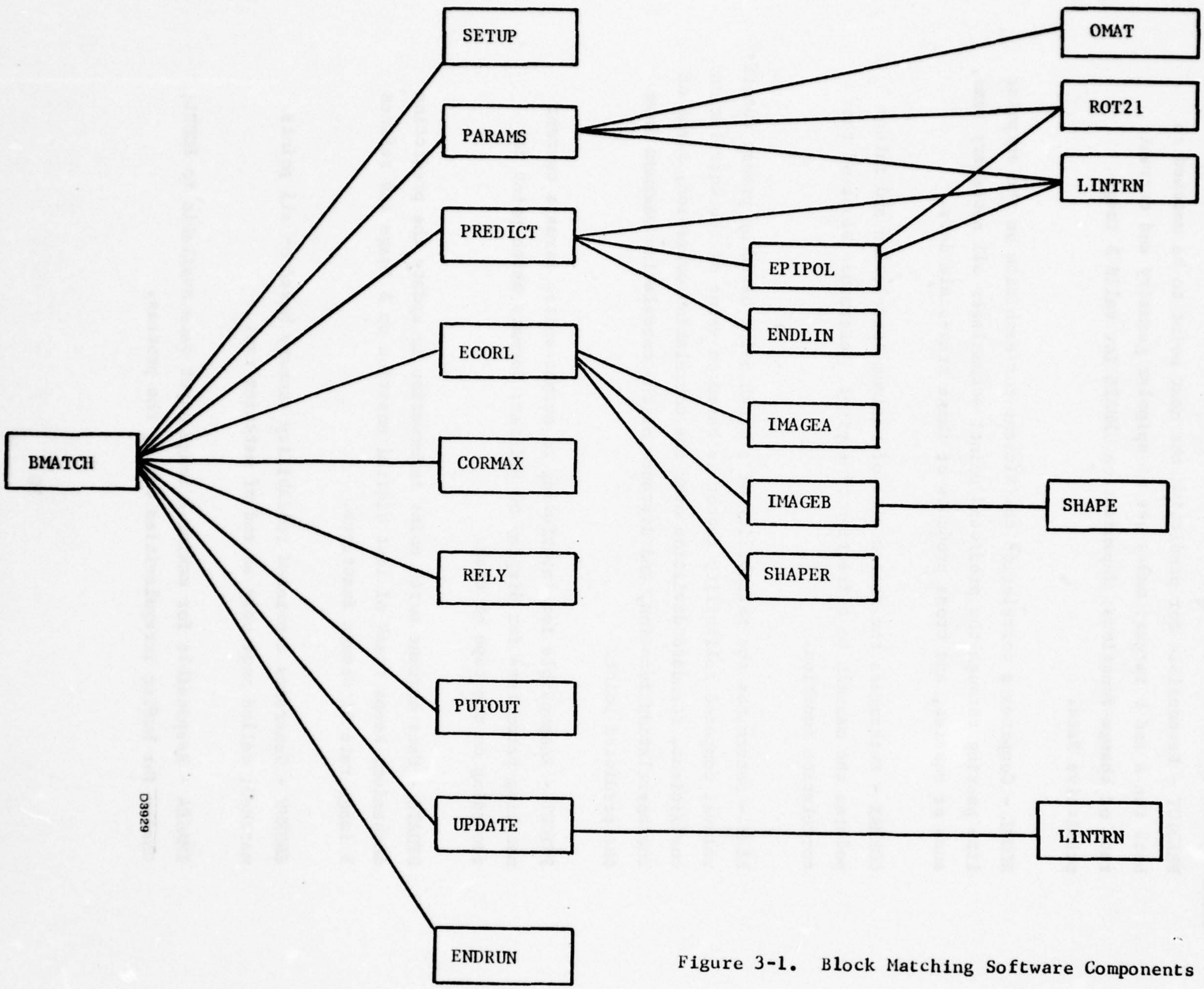


Figure 3-1. Block Matching Software Components

PREDICT - Responsible for predicting the next point to be matched on both the A and B images; makes use of epipolar geometry and current rate of change functions; dependent on UPDATE for valid B image predictive data.

ECORL - Computes a correlation coefficient for each site on the epipolar line passing through the predicted point; accumulates all necessary sums, sums of squares, and cross products of image gray-scale data.

CORMAX - Determines the search site of maximum correlation and interpolates the maximum to a fraction of a pixel; determines slope of the correlation function.

RELY - Determines the reliability of a match point based on preset cutoff values; computed reliability factor is based on value of the correlation coefficient, standard deviation over the correlation subregion, slope of the correlation function, and distance of the correlation maximum from the predicted point.

PUTOUT - Responsible for interfacing to output units; formats current matching parameters for display on printer; formats match point for recording on mag tape or disk.

UPDATE - Uses current match point information to update the prediction mechanism; keeps track of last digital position on B image and refines B image rate of change functions.

ENDRUN - Generates a printed reliability summary based on all points matched; called once only at end of matching run.

IMAGEA - Responsible for making A image ouput data available to ECORL; checks for buffer irregularities during the process.

IMAGEB - Responsible for making B image input data available to SHAPER; checks for buffer irregularities during the process.

SHAPE - Computes B image patch shaping parameters based on current size and rate of change functions in the vicinity of the patch.

SHAPER - Generates one row of shaped B image data for use by ECORL; uses parameters computed by SHAPE; called once for every pixel row of correlation patch.

EPIPOL - Computes the coefficients of corresponding epipolar lines on the A and B images passing through the predicted point; coefficients used by PREDICT.

ENDLIN - Checks for wandering blocks in a complete line of match points (all points with the same x coordinate on the A image); corrects any blocks found wandering; called only when a line of match points is complete.

OMAT - Computes the 3 x 3 orientation matrix from the exposure tilt angles Ω ϕ, K; called once only by PARAMS.

ROT21 - Performs the rotation of a point in one coordinate system into another coordinate system on the basis of a 3 x 3 orientation matrix.

LINTRN - Performs the linear transformation of a point from one two-dimensional coordinate system into another; used for transforming digital scan coordinates to photo coordinates and vice versa.

As a further explanation of the process, BMATCH contains only a control loop that calls the necessary components into operation in the proper sequence. As will be seen later, BMATCH simulates the operation of a host computer in sequencing the operation of other processors. The actual component sequence is shown in Figure 3-2. It is set up such that one execution of the control loop produces one match point.

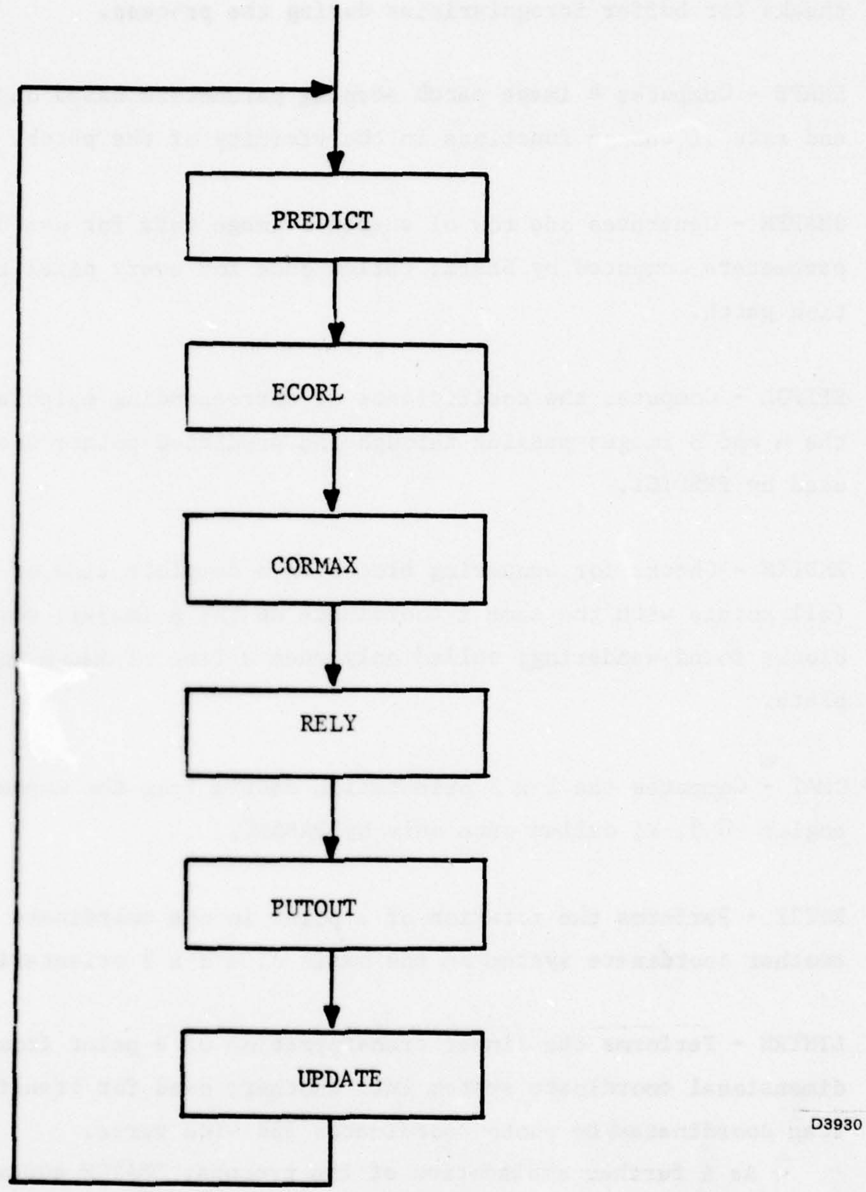


Figure 3-2. Primary Matching Loop (BMATCH)

3.2 DATA ORGANIZATION

Just as the matching algorithm in terms of process control has been organized into logical components, so too the data and parameters operated on by the algorithm have been organized into discrete packets. These packets are used as communication links between the various process components. Figure 3-3 shows this data communication and Figure 3-4 is a table showing all of the variables in each packet and the occurrence of each packet over the collection of logical components. A star indicates that the indicated component has a need for the indicated packet. A description of what each packet contains follows.

GRID - Starting and ending positions and increments for the matching grid over the desired area of processing.

ABUF - Image A gray-scale data and buffer management parameters.

ACOR - Match point coordinates on image A in digital scan system and photo system and associated parameters.

BBUF - Image B gray-scale data and buffer management parameters.

BCOR - Match point coordinates on image B in digital scan system and photo system and associated parameters.

COR - Correlation search area parameters; values of correlation coefficient, standard deviation, correlation slope, and deviation from predicted point.

PRED - Rate of change prediction functions for each path of blocks.

IO - Interior orientation transformations in X and Y for image A and B to convert digital scan coordinates to data coordinates and vice versa.

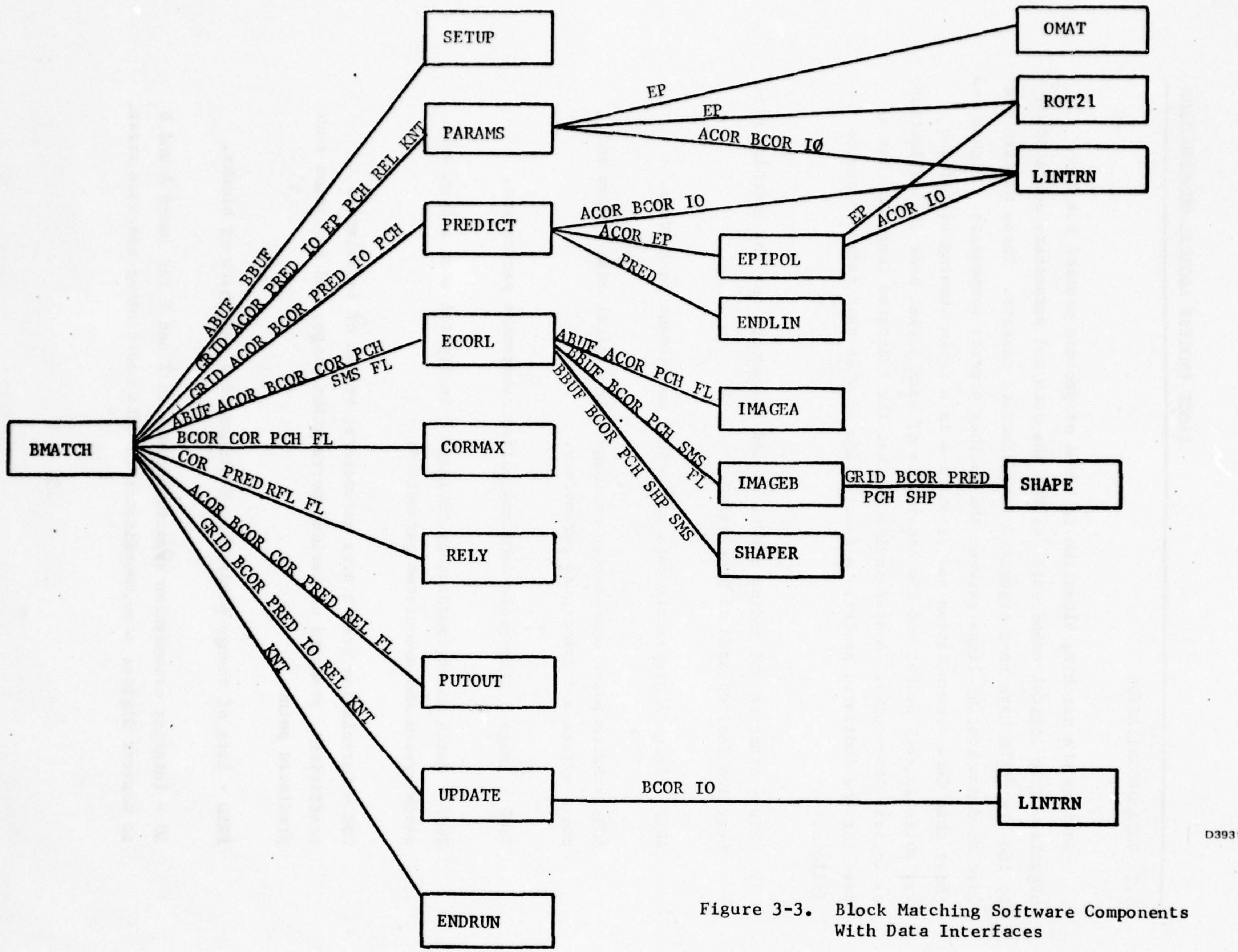


Figure 3-3. Block Matching Software Components With Data Interfaces

	GRID	ABUF	ACOR	BBUF	BCOR	COR	PRED	IO	EP	PCH	SHP	SMS	REL	KNT	FL
	PSTI PNDI PDX PSTJ PNDJ PDY	IMA() IALEN NWA JALEN ILA ILNA JPA JPNA IABP	PA() DPA() LAST IWA ICHA NA MA	IMB() IBLEN NWB JBLEN ILB ILNB JPB JPNB IBBP	PB() DPB() DPBR IBST IWB ICHB NB MB	CORC() LS SDA CPK CSL UDEV	PBR() PBRD() PDU() KP KPN	FXA() FYA() FXIA() FYIA() FXB() FYB() FXIB() FYIB()	BASEB() ORMBA() EA EB	HXA HYA SX SHF	ISH LST LND SH() DSH() RK() DRK() IRK INDX	IPB() SUMB() SSQB() SXY()	TCC TSD TDU TSL IRV() IRFAC	NPTS IRTAB()	IFLA IFLB IFLC
BMATCH	*		*												
PARAMS	*		*				*	*	*	*			*	*	
SETUP		*		*											
PREDICT	*		*		*		*	*	*						
ENDLIN							*								
ECORL		*	*		*	*				*		*			*
SHAPER				*	*					*	*	*			
CORMAX					*	*				*					*
RELY						*	*						*		*
UPDATE	*				*		*	*					*	*	
IMAGEA		*	*							*					*
IMAGEB				*	*					*	*	*			*
SHAPE	*				*		*			*	*				
PUTOUT			*		*	*	*						*		*
EPIPOL			*						*						
ENDRUN														*	

Figure 3-4. Block Matching Data Interfaces

EP - Relative orientation parameters and epipolar line coefficients.

PCH - Correlation patch size and shape and search segment length parameters.

SHP - B image correlation patch shaping functions and differential increments.

SMS - Correlation coefficient sums and cross product accumulators.

RFL - Reliability factor and reliability criteria cut off values.

KNT - Reliability frequency accumulators.

FL - Process error flags.

The attempt is being made to maintain the integrity of these data packets as well as the integrity of each logical component as the process is reconstructed in microcode. This type of modularization is the key to straightforward management and subsequent modification of the software structure.

4.0 BLOCK PROCESS RECONSTRUCTION

The actual benchmark of the block matching algorithm will take place on a configuration of microprogrammable processors called the Flexible Processor (FP) Array. This is an experimental configuration being constructed by the Digital Image Systems Division for image processing research and demonstration purposes. The configuration contains four Flexible Processors with their associated memory modules controlled by a CDC 1700 computer. This configuration is shown in Figure 4-1. Also shown as part of the configuration is a data channel controller (DCC) with 32K of memory. This is called a line buffer memory and is used to interface image data at very high speeds to and from FPs and peripheral devices. A fifth Flexible Processor is included in the configuration as part of a digtal scan converter. The role of this FP is to manage the large display memory and to constantly refresh a 512 by 640 element color CRT.

The Flexible Processors in this configuration are all uniform in design and connected together in a pipeline fashion through the standard AQ and DSA channels. In fact, the attempt is being made to construct the entire configuration in what can be considered a standardized fashion. This is the key to being able to implement many different algorithms and applications on the same hardware configuration. This represents a different approach from the FP configurations that have been constructed in the past. In such systems, one of which is the Four Channel Modular Change Detection System, the Flexible Processors in the processing pipeline can be different from one another, having different amounts of micromemory and different communication paths. In short, the configuration is tailored to meet the real-time needs of a truly special purpose processing task. It is difficult to implement a variety of algorithms on such a configuration.

Flexible Processors as well as banks of MOS memory are all designed as modular hardware building blocks. As such, large numbers of them can be cabled together to produce a processor of considerable computational power. If the

03932

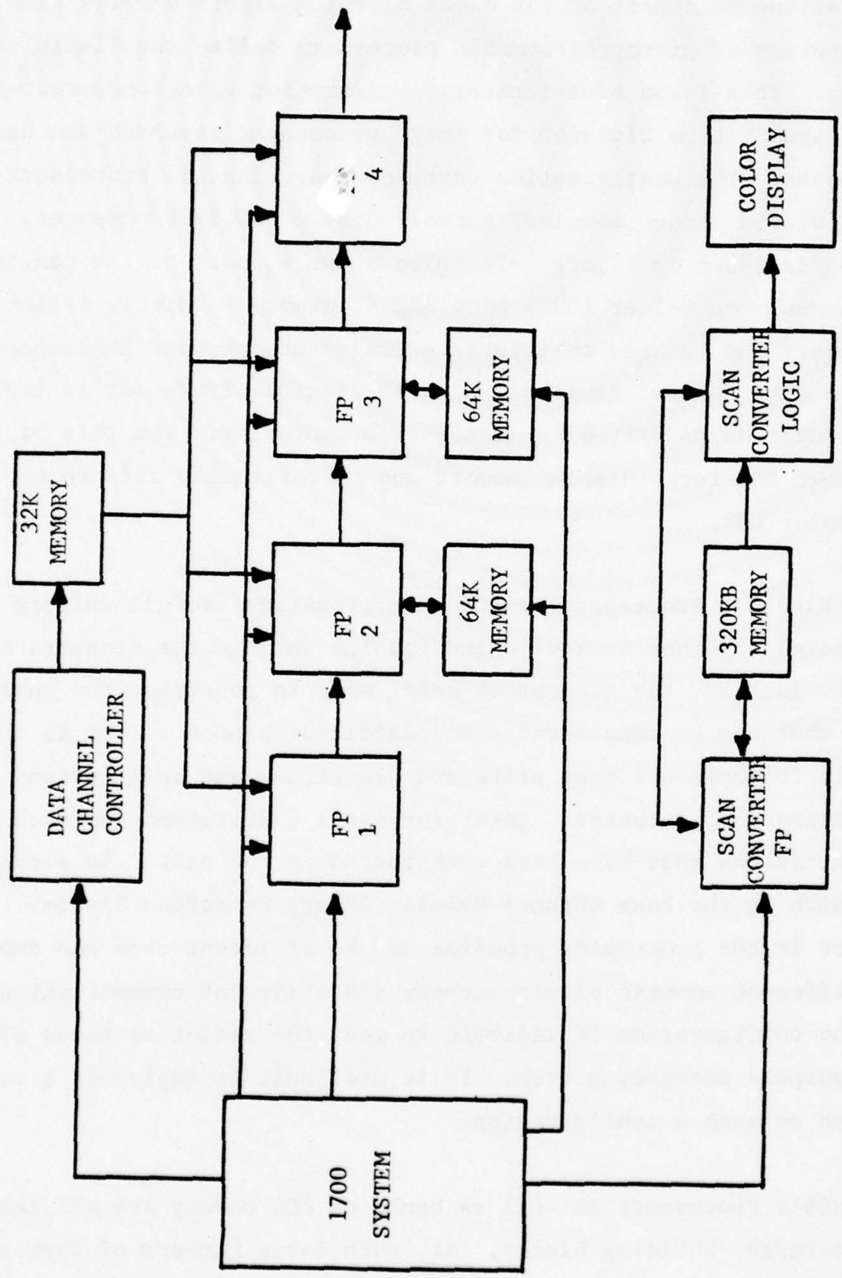


Figure 4-1. FP Array Configuration

construction of the FPs and memory is uniform and the cabling between them standardized, then the resulting configuration is also very flexible as well as expandable.

Figure 4-2 illustrates this concept of expandability. Typically, an algorithm is implemented on a collection of FPs that are connected in a pipeline fashion such that the first FP receives some data and does some computations on it and then passes the results to the next FP along the pipeline. While the second FP is doing its portion of the processing and passing results to the next, the first is processing new data, and so on through the pipe. In this way, processing parallelism is achieved because each FP performs its portion of the algorithm simultaneously with the others.

Such a pipeline of FPs can be considered as one processing channel. To provide more computational power or higher data throughput, more FPs can be added to the pipeline up to a total of 16 per channel. Also as another dimension to the configuration expandability, a channel may be duplicated up to 16 times in the direction shown in the figure. Particularly in image processing, this channel duplication can result in much increased system throughput because the image data can be partitioned such that each channel runs the same algorithm in parallel on its portion of the data. Thus, a four channel configuration can theoretically process the same data as a one channel system in one fourth the time.

However, duplicate hardware channels need not run the same algorithm. Some channels can be loaded with microcode to perform completely independent tasks on different sets of data. In this sense a multiple channel FP configuration can be truly classed as a multiprocessing system.

The above discussion has been provided as background to show that the current benchmark is being performed as a demonstration of the capability of a hardware subset that can evolve into a flexible and powerful system.

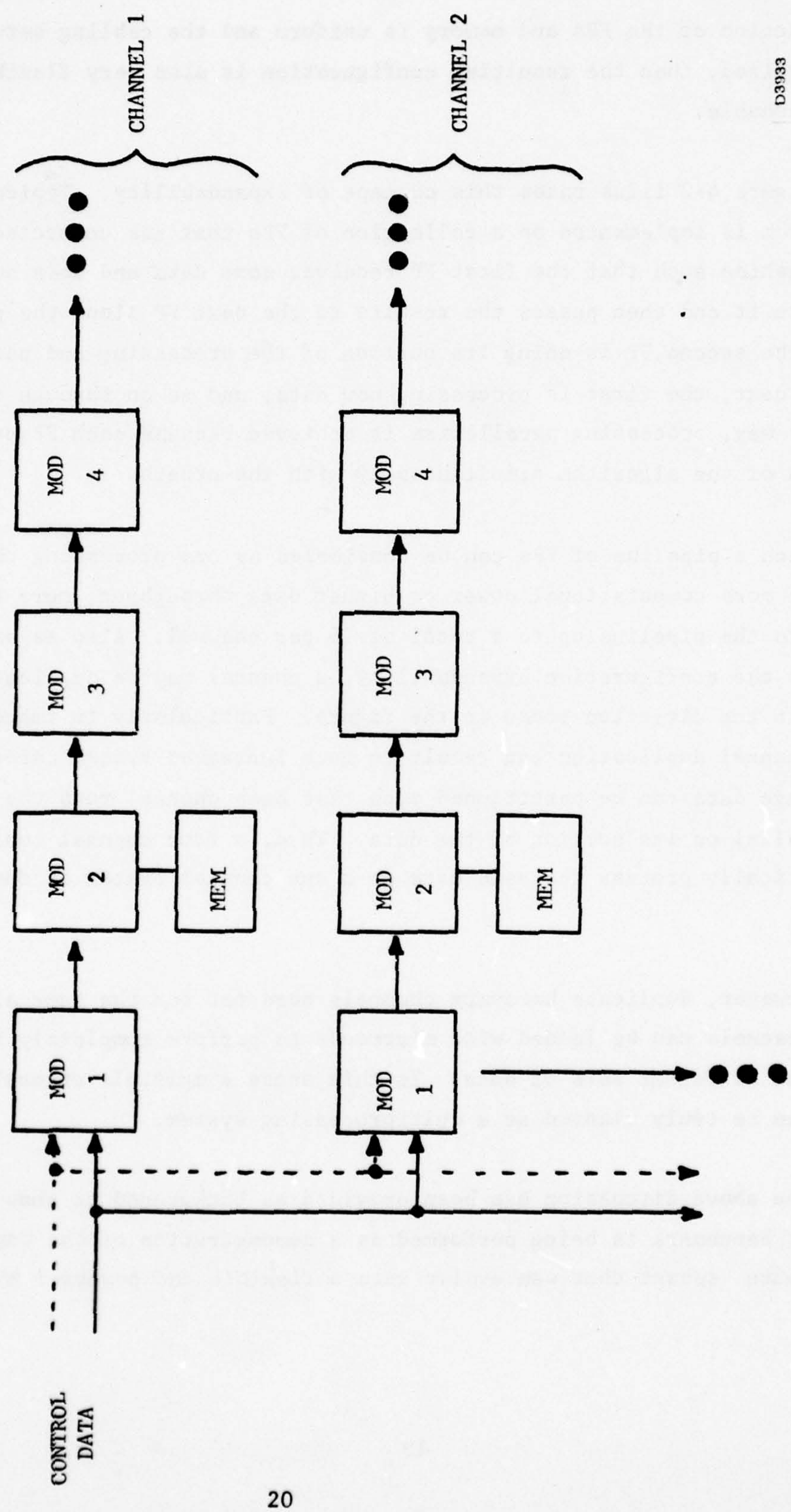


Figure 4-2. Configuration Expandability

4.1 ALGORITHM DISTRIBUTION

What is available now to run the benchmark is a configuration of four Flexible Processors arranged in one channel. In the logic level reconstruction that follows Flexible Processors 1, 2, 3, and 4 of Figure 4-1 will be referred as modules 1, 2, 3, and 4 respectively. The CDC 1700 control computer is the logical host. Implementation of the block matching algorithm on the available hardware then involves the distribution of the logical components of the algorithm over the host and four microcode modules.

Figure 4-3 illustrates this distribution. The host module contains the basic process control loop BMATCH and the initialization components SETUP and PARAMS. ENDLIN has been included as part of BMATCH and is not shown because of its minimal function. The components ØMAT, RØT21, and LINTRN have likewise been included as part of their calling components.

The theory of operation of the resulting pipeline is as follows. The host module is responsible for loading the proper microcode, sequencing the process, supplying data to the large MOS memory, and depositing the output match points on disk. MOD1 computes the predicted match point locations for an entire line of blocks and passes this information to MOD2. It then receives refined match point data from MOD4 and starts the prediction for the next line of blocks. Meanwhile MOD2 uses the data received from MOD1 to compute the patch shaping functions and to start accessing image data from memory. As soon as one patch row of data is shaped, it is sent to MOD3 for statistical accumulation and the next patch row is shaped. As this is occuring, MOD3 accumulates the necessary partial sums for the next row. For each row of the patch MOD2 and MOD3 are constantly talking to one another. When MOD3 finishes the last row of the patch, it sends the correlation data to MOD4 which determines the maximum and reliability factor and prepares to output the match point data to the host and MOD1. At this time, MOD2 has gone on to the next block. This goes on in feedback fashion until the host has determined that the process is complete.

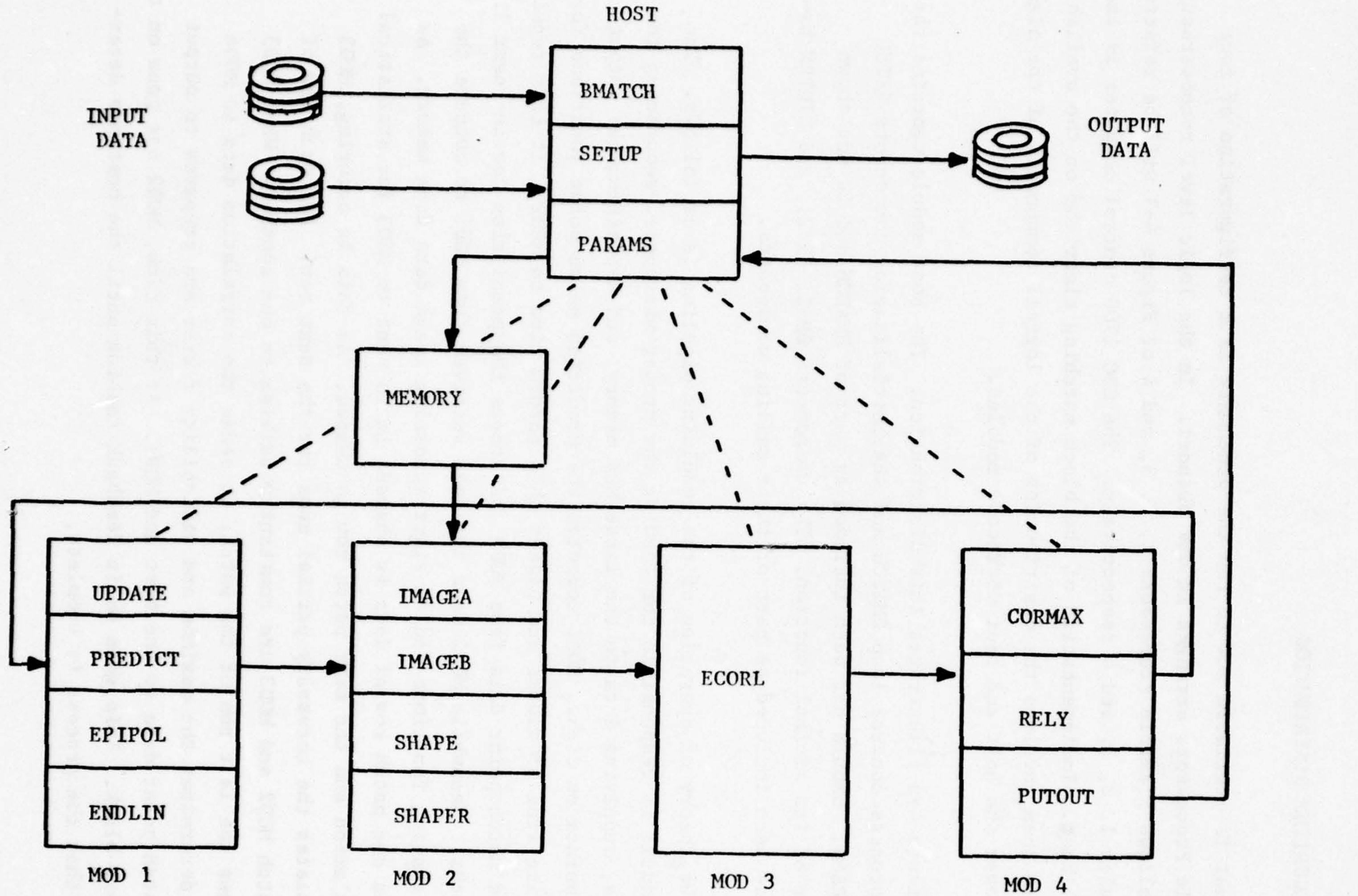


Figure 4-3. Block Matching Microprocessor Implementation

In breaking apart each of the processing modules, it has been found that the relative computational complexity in equivalent add operations per match point is as follows:

MOD1	159
MOD2	5851
MOD3	23874
MOD4	143

It can be seen that MOD3 is the workhorse of the process.

The above represents a rather conservative distribution of the algorithm's compute load. Theoretically, in a distributive computing network such as the FP array, one should be able to take the total number of operations or the total compute load of the algorithm and divide it by four, thus evenly distributing the load among the four processing modules. The problem is that some components contain many microcode statements to perform a few operations whereas other components contain few microcode statements in tight loops that execute many operations. ECORL is a case of the latter. Each FP has but 1K words of micromemory, although many operations can be performed in one microinstruction. Therefore, the problem of redistributing the above compute load is dependent upon how many logical components can fit in each module's micromemory.

An alternative implementation which incorporates a different compute load distribution is shown in Figure 4-4. The relative computational complexity is as follows:

MOD1	903
MOD2	7812
MOD3	21669
MOD4	143

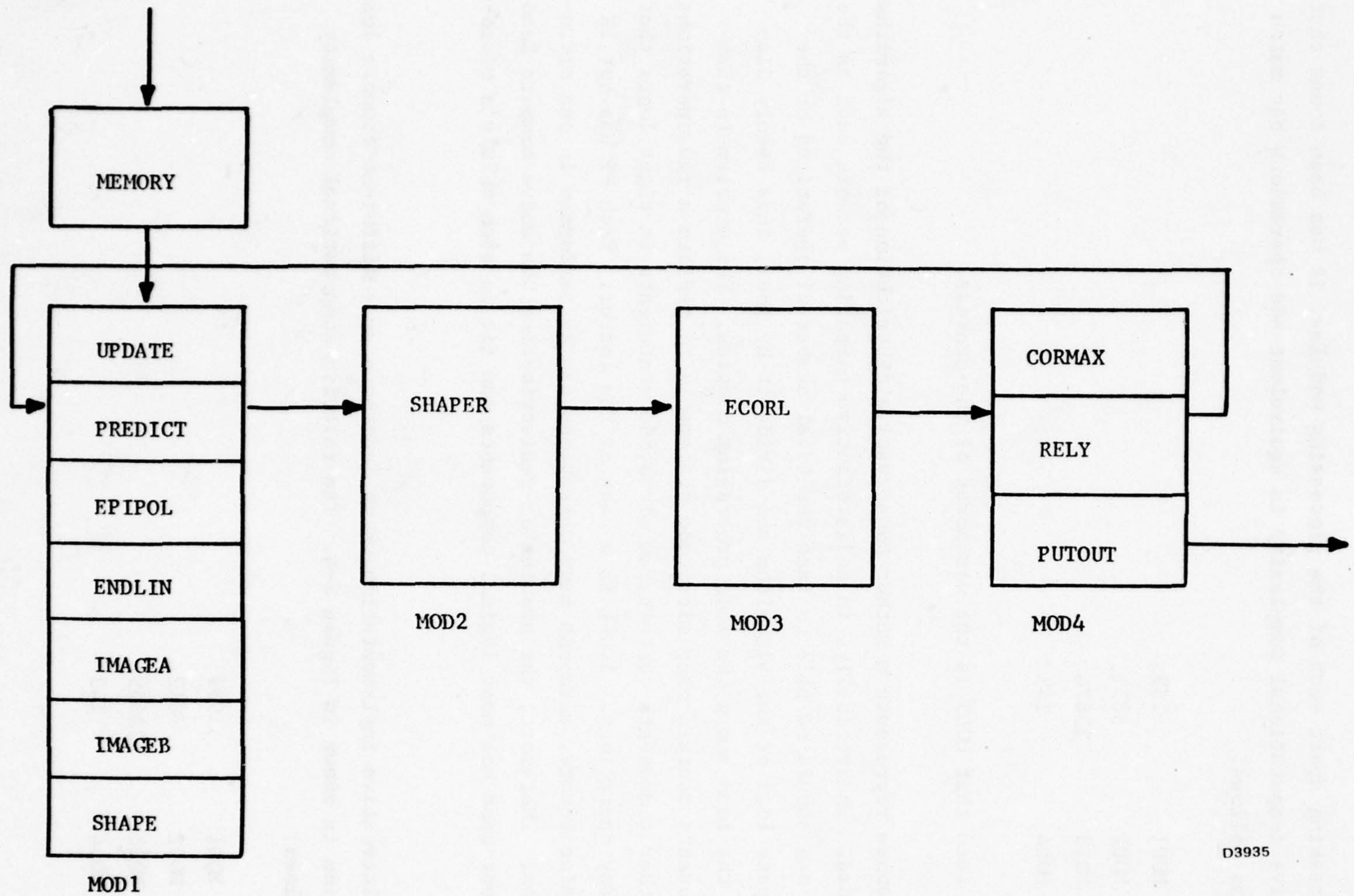


Figure 4-4. Alternate Implementation

Here, some of the responsibility for accumulating gray scale sums has been taken from ECORL in MOD3 and incorporated into SHAPER in MOD2. Also, the image data accessing components and the shaping computations have been pushed back to MOD1. MOD4 has been left relatively unloaded because it must spend a lot of time communicating with the host as regards output data, a slow process in comparison to the extremely fast computations of the other modules.

These implementations have been provided as initial estimates to size up the microcoding task. Further redistribution is possible when the microprogramming is actually performed. One area to be considered is whether some of the responsibility of MOD3 can be placed in MOD4. Another aspect to be firmed up during actual microprogramming is how to obtain the optimum process parallelism such that the time spent by a module waiting for another module is minimized.

5.0 BLOCK PROCESS TIMING ESTIMATE

In determining how fast the benchmark configuration will actually execute the block matching algorithm over a stereo model area, the following points are relevant:

- Matching occurs for approximately 12000 points per square inch of image.
- A typical stereo overlap is 54 square inches.
- Therfore, 648000 points per model area are generated.
- In a general-purpose environment, where the block matching algorithm has a rate of 14 points per second, a model area would take approximately 13 hours of processing time.
- In the benchmark configuration, the processing time is dependent on the number of operations per point required by the most heavily loaded processor in the configuration, provided that the optimum parallelism has been achieved.
- The time required by a Flexible Processor to perform one add operation is 125 nanoseconds.

With these considerations in mind, the benchmark timing should fall within the range of the following table, depending on the equivalent number of sequential operations performed by the most heavily loaded processor.

OPERATIONS PER MATCH POINT	SECOND PER MATCH POINT	TOTAL MODEL TIME IN MINUTES
10000	.00125	13.5
15000	.00188	20.3
20000	.00250	27.0
25000	.00313	33.8
30000	.00375	40.5
35000	.00438	47.3

These times have been derived in accordance with matching data where the image pixel side covers four feet on the ground, the correlation patch size is 21 by 21 pixels and the match point grid interval is ten pixels by eight lines. This is a somewhat typical matching situation. But the times are subject to change with different image data depending on the scale of the imagery, density of matching and correlation patch and search segment size.

6.0 CONCLUSION

The block matching algorithm has been logically reconstructed and is ready for microprogramming. A number of problems involving the optimum distribution of logical components and the size of the available micromemories have been uncovered in Phase B which should be resolved in Phase C during microprogramming. The attempt during Phase B has been to keep the development on a rather high logical level, deferring discussions of communication and data channels, instruction formats, and I/O transfer rates to the microprogramming task of Phase C.

DIGITAL CARTOGRAPHIC STUDY AND BENCHMARK

THIRD INTERIM TECHNICAL REPORT

Prepared for:

U. S. Army Engineer Topographic Laboratories

Fort Belvoir, Virginia

Contract DAAG53-75-C-0195

Prepared by:

D. J. Panton

M. E. Murphy

September 1976

Control Data Corporation

DIGITAL IMAGE SYSTEMS DIVISION

2800 East Old Shakopee Road

Minneapolis, Minnesota 55420

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1.0 INTRODUCTION

This report describes the work and findings of Phase C of the Digital Cartographic Study and Benchmark. The primary purpose of Phase C was to implement and test on fast, microprogrammable processors the stereo matching algorithm that was designed and analyzed under Phases A and B of the program.

This implementation was performed in terms of a benchmark to shed some light on the practicality of such a concept and to uncover the advantages and disadvantages of this particular kind of parallel processing application. The results stem from actually performing the benchmark rather than from a paper study outlining the approach to be taken.

The complexity of the benchmark algorithm is representative of today's advanced digital image processing applications. That is, the algorithm is complex to the extent that production implementation on a general-purpose computer system is not practical. The benchmark hardware is of a state-of-the-art parallel processing nature with fast instruction cycle times and complete microprogramming capability. The numerical results are as accurate as the simulation results of Phase A and the benchmark speeds are well within the range predicted in Phase B.

1.1 SUMMARY OF RESULTS

The benchmark was evaluated both in terms of its speed and accuracy. The matching algorithm when implemented on a CDC 6600 computer, a large general-purpose system, generates 14 match points per second for an average set of matching parameters. For the same test case, the benchmark algorithm generates 270 match points per second. Thus, the benchmark implementation runs approximately 19 times faster. Compared with the slower general-purpose system, the CDC 6400, which produces about eight match points per second, the benchmark implementation is approximately 34 times faster. Stated another way, it takes about 11 hours of CDC 6600 processing to complete the correlation and matching

of a typical 9 x 9 inch frame stereo overlap area. The benchmark implementation can process the same area in a little over one half hour.

In terms of accuracy, the algorithm simulator that runs on the CDC 6600 and CDC 6400 performs arithmetic operations using the 60-bit word available on these machines. Floating point operands consist of 12 bits of exponent and 48 bits of mantissa. Computation accuracy with this word length is unquestioned; any inaccuracy here is attributed to algorithm design rather than the computing system used. However, the benchmark configuration is composed of 16-bit machines. The benchmark algorithm uses two words, or 32 bits, for most operands. But despite the smaller word length, the benchmark results were comparable in accuracy to the simulator results.

2.0 BENCHMARK SYSTEM COMPONENTS

The purpose of this section is to list the components of the benchmark system as a framework for the discussions that follow.

Hardware Components:

1) A CDC 1700 computer with the following peripherals:

   - tape controller and three magnetic tape units
   - disk controller and operating system disk drive
   - character display and keyboard for user interaction

2) Four CDC Flexible Processors with associated data channels

Microcode Components:

1) Algorithm microprograms for each Flexible Processor

2) A LYNK package in one Flexible Processor

3) Special interrupt service microprograms in all FP's to handle specialized communication

Software Components:

1) COR operating system to run on the 1700 computer

2) A microcode assembler

3) A file system for creating and editing microprograms and command files

4) One overlay in the operating system to handle matching algorithm control functions

5) A set of hardware diagnostic programs to check the integrity of Flexible Processor components

6) Utility programs to dump FP register files and to perform various debugging operations.

3.0 BENCHMARK HARDWARE DESCRIPTION

The benchmark hardware configuration contains two basic computational units: a host computer that performs control functions and communicates with the external environment and an array of parallel processors that execute the image processing algorithm. The host computer is the CDC 1700, a small scale minicomputer that is typically used for production control applications. The 1700 is a 16-bit one's complement machine with an instruction cycle time of 1.1 microsecond.

The majority of computing for the benchmark is performed by the array of microprogrammable processors, each of which is a CDC Flexible Processor (FP). The FP has been designed as a modular hardware building block that can be configured in arrays to perform a wide variety of image processing applications. Each Flexible Processor is a microprogrammable computing unit that features high arithmetic computation rates and high data throughput rates. Additional FP characteristics are listed in Table 3-1.

Internally, the Flexible Processor is organized in a dual bus architecture. Figure 3-1 illustrates this organization. The register files shown in the figure are 60-nanosecond semiconductor memories that have simultaneous read/write capability. They are shown as having a word size of 32 bits, but in actual benchmark implementation they are considered as being double files of 16-bit words. For example, in the benchmark configuration each FP has two large files each containing 2048 16-bit words.

There are three basic types of communication channels that can be used to connect Flexible Processors. The A/Q channel is compatible with CDC 1700 equipment and is used primarily by FP's for single word transfers of control information. The DSA channel is a data channel that is faster than the A/Q channel and is used for accessing data from and storing data into an external MOS semiconductor memory. The third type of channel is the high-speed channel used for interprocessor data communication. The transfer rate on this channel

is eight megawords per second, making it ideal for transferring image data from FP to FP.

TABLE 3-1. CDC FLEXIBLE PROCESSOR CHARACTERISTICS

- Microprogrammable-Random Access Microcontrol Memory
- 32-Bit or 16-Bit Word Lengths
- Array Hardware Multiplier
- 16-Level Hardware Priority Interrupt Mechanism - Three-Level Mask Capability
- Specialized Logic for Square Root and Divide
- 8 MHz File Buffered Word Transfer Rate - 16-Word by 32-Bit or 16-Bit Input File Buffer
- 2 MHz Direct Memory Access Word Transfer Rate
- 1 MHz Register-Buffered Word Transfer Rates
- Dual 16-Bit Internal Data Bus System
- 0.125 μs Clock Cycle
- 0.125 μs 32-Bit Addition: 0.250 μs Byte Multiplication
- Register File Capacity Up to 4128 16-Bit Words
- Hardware Network for Conditional Microinstruction Execution - Four Mask Registers and a Condition Hold Register

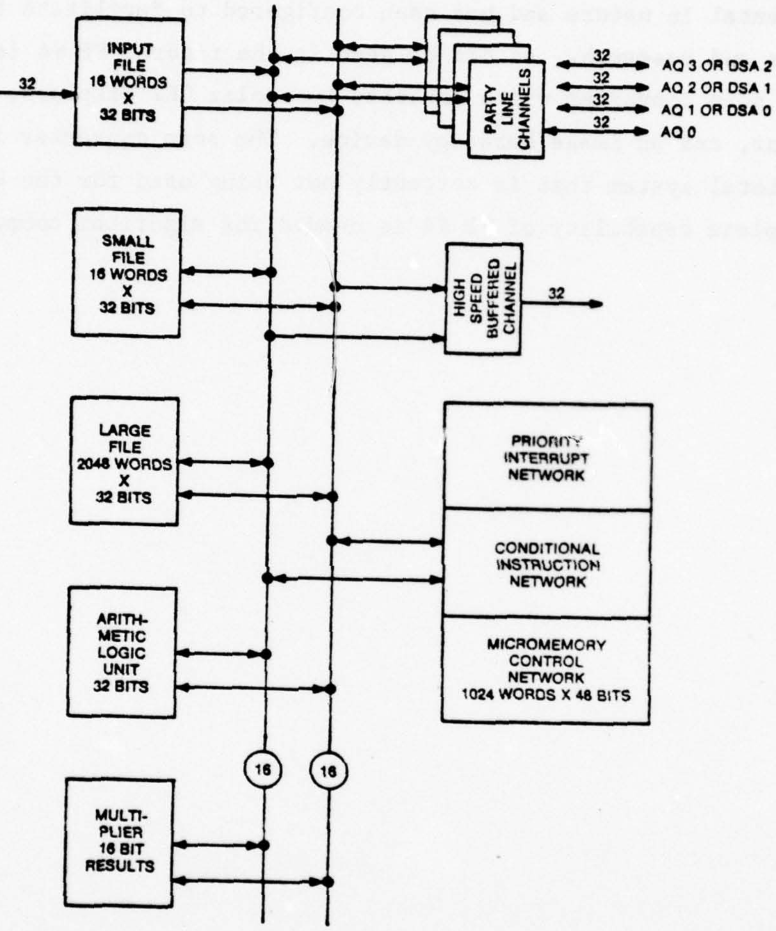


Figure 3-1. Internal Organization of the CDC Flexible Processor

The entire system on which the benchmark was implemented is shown in Figure 3-2. This system has been assembled in the laboratory of the Digital Image Systems Division and has been evolving for several years. It is experimental in nature and has been configured to facilitate interactive image analysis and research. As can be seen in the figure, FP #4 is used to drive a digital scan converter which controls two color CRT displays, a color image projector, and an image hardcopy device. The scan converter is one resource of the total system that is currently not being used for the benchmark because the complete capability of FP #4 is needed for algorithm computations.

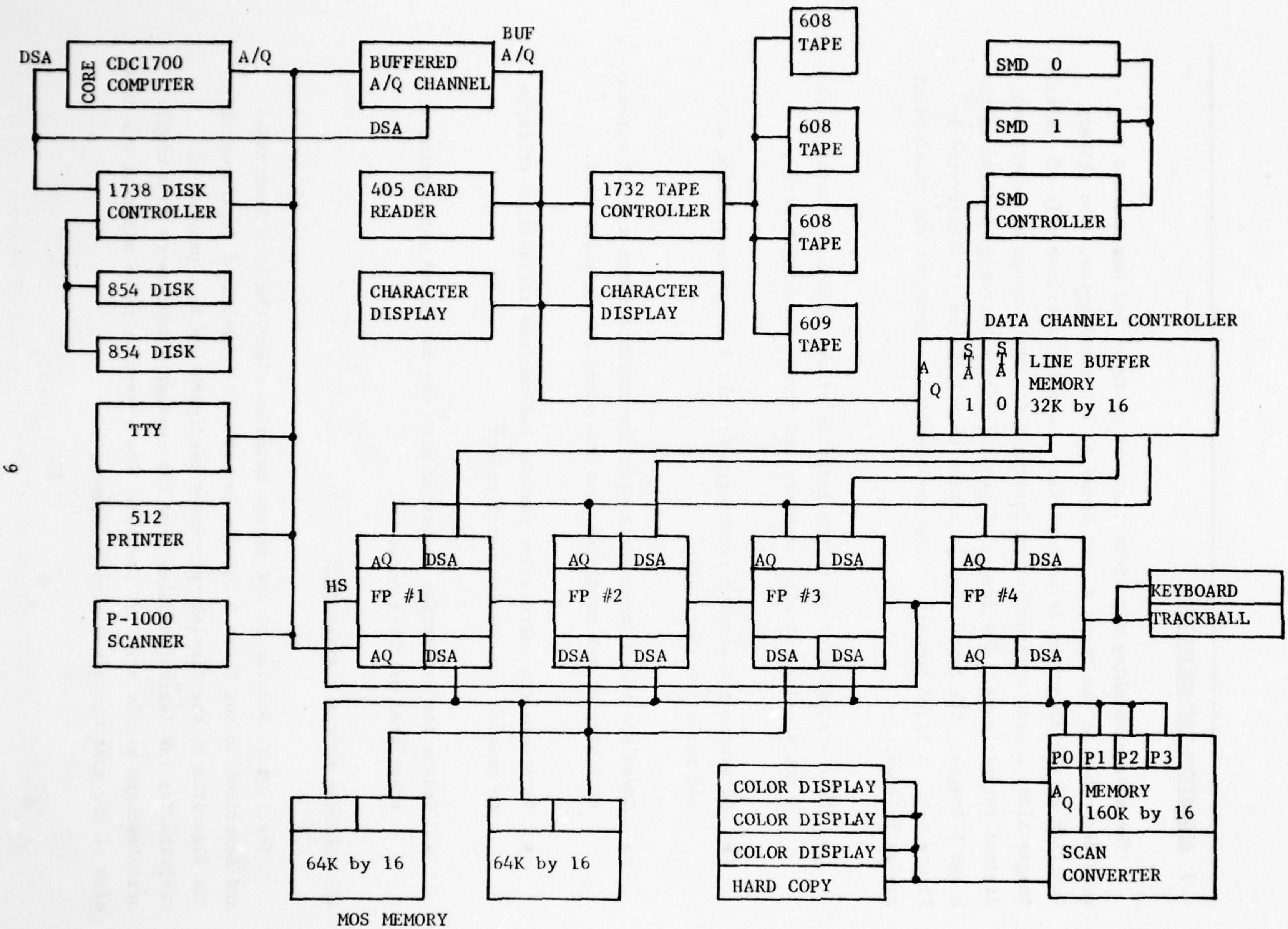


Figure 3-2. Total Benchmark Configuration

D3936

4.0 ALGORITHM IMPLEMENTATION

The stereo matching algorithm that was designed and benchmarked is well-described in the Phase A, and Phase B, Interim Reports. The primary function of the algorithm is to correlate and match a stereo pair of digital images along a grid with arbitrary intervals for the purpose of generating digital terrain data. The images of the pair are arbitrarily denoted as the A and B images. The basic concept behind the algorithm is illustrated in Figure 4-1. A brief summary of the steps needed to generate one match point are as follows:

- Make a prediction as to the location of the next match point on the B image using epipolar geometry and recent matching history.
- Determine the shaping parameters for the B image correlation area and shape the image data.
- Correlate the A image patch with the shaped B image area at sites on each side of the predicted location along the epipolar line.
- Determine the correlation maximum and various reliability criteria to compute the precise match point.
- Update the prediction mechanism with the new found match point and associated matching data.

4.1 LOGICAL MODULE DISTRIBUTION

The logical modules of the stereo matching algorithm have been named and described in the Phase B Interim Report. The problem of implementing the algorithm on the flexible processor configuration is a problem of distributing the logical modules over the available processors in a highly optimized and highly parallel fashion. The objective is to minimize the idle time of the most heavily loaded processor.

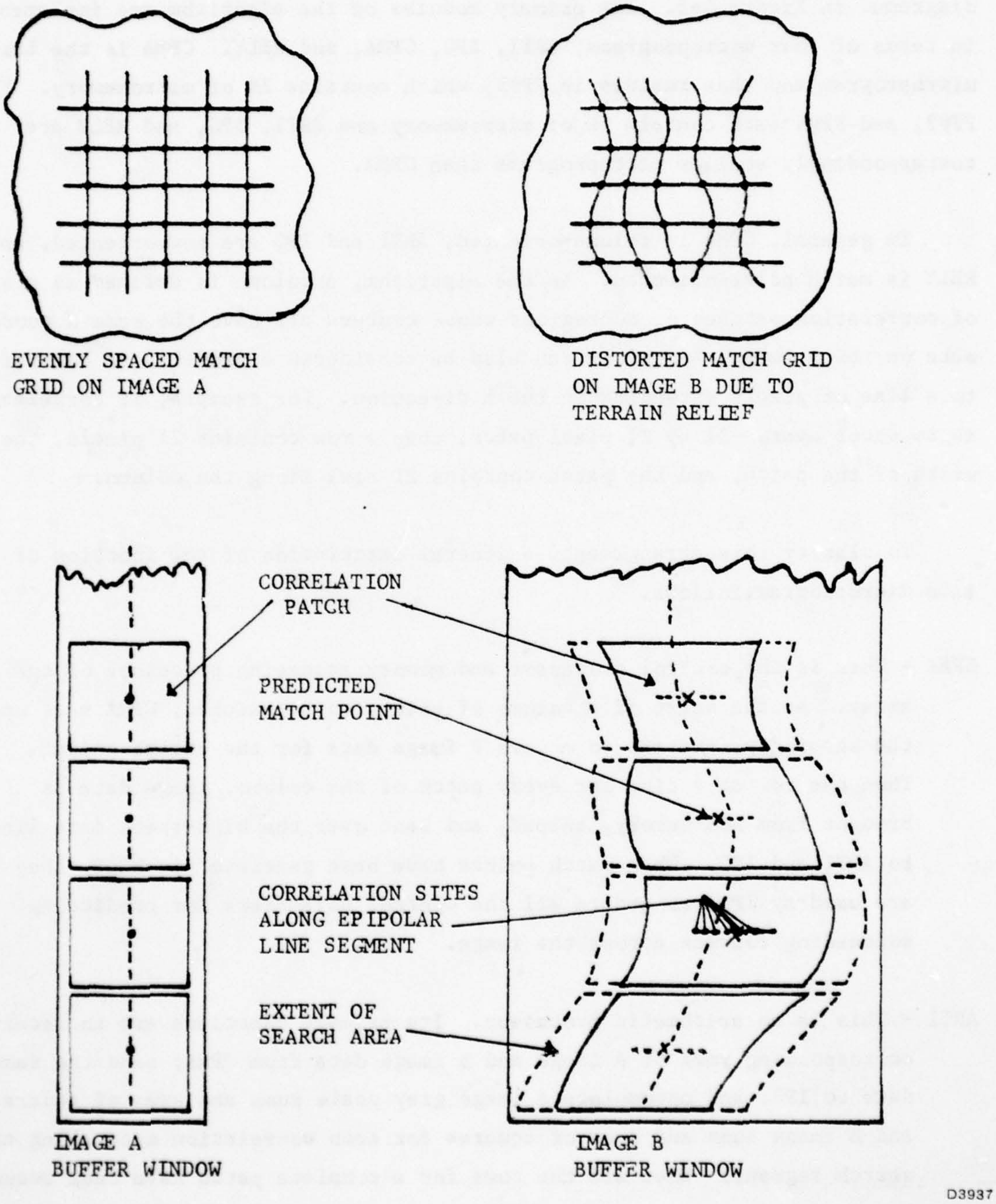


Figure 4-1. Block Matching Conceptualization

The final distribution of logical modules that was decided upon is diagramed in Figure 4-2. The primary modules of the algorithm are implemented in terms of four microprograms, ART1, IPC, CPMA, and RELY. CPMA is the largest microprogram and thus resides in FP#3, which contains 2K of micromemory. FP#1, FP#2, and FP#4 each contain 1K of micromemory and ART1, IPC, and RELY are correspondingly smaller microprograms than CPMA.

In general, CPMA is column-oriented, ART1 and IPC are row-oriented, and RELY is match point-oriented. In the algorithm, a column is defined as a set of correlation patches or subregions whose centers all have the same X coordinate on the A image. A column can also be considered a profile. A row refers to a line of pixels extending in the X direction. For example, if correlation is to occur over a 21 by 21 pixel patch, then a row contains 21 pixels, the width of the patch, and the patch contains 21 rows along the column.

To clarify this arrangement, a general description of the function of each microprogram follows.

CPMA - This is the control processor and memory accessing processor of the array. At the start of a column of points to be matched, CPMA sets up the shaping parameters to access B image data for the entire column. Then one row at a time for every patch of the column, image data is brought from MOS memory, shaped, and sent over the high-speed data line to ART1 and IPC. When match points have been generated in RELY, they are used by CPMA to update all the control parameters for predicting succeeding columns across the image.

ART1 - This is an arithmetic processor. Its primary functions are to receive corresponding rows of A image and B image data from CPMA; send the same data to IPC; and accumulate A image gray scale sums and sums of squares and B image sums and sums of squares for each correlation site along the search segment. When all the rows for a complete patch have been accumulated, the sums are sent to RELY. One execution of the main loop of ART1 accumulates one row of pixels.

AD-A064 800 CONTROL DATA CORP MINNEAPOLIS MINN DIGITAL IMAGE SYS--ETC F/G 8/2
DIGITAL CARTOGRAPHIC STUDY AND BENCHMARK.(U)
DEC 78 D J PANTON, M E MURPHY, D S HANSON DAAG53-75-C-0195
UNCLASSIFIED CDC-76003-78-R6 ETL-0168 NL

2 OF 3
ADA
064800

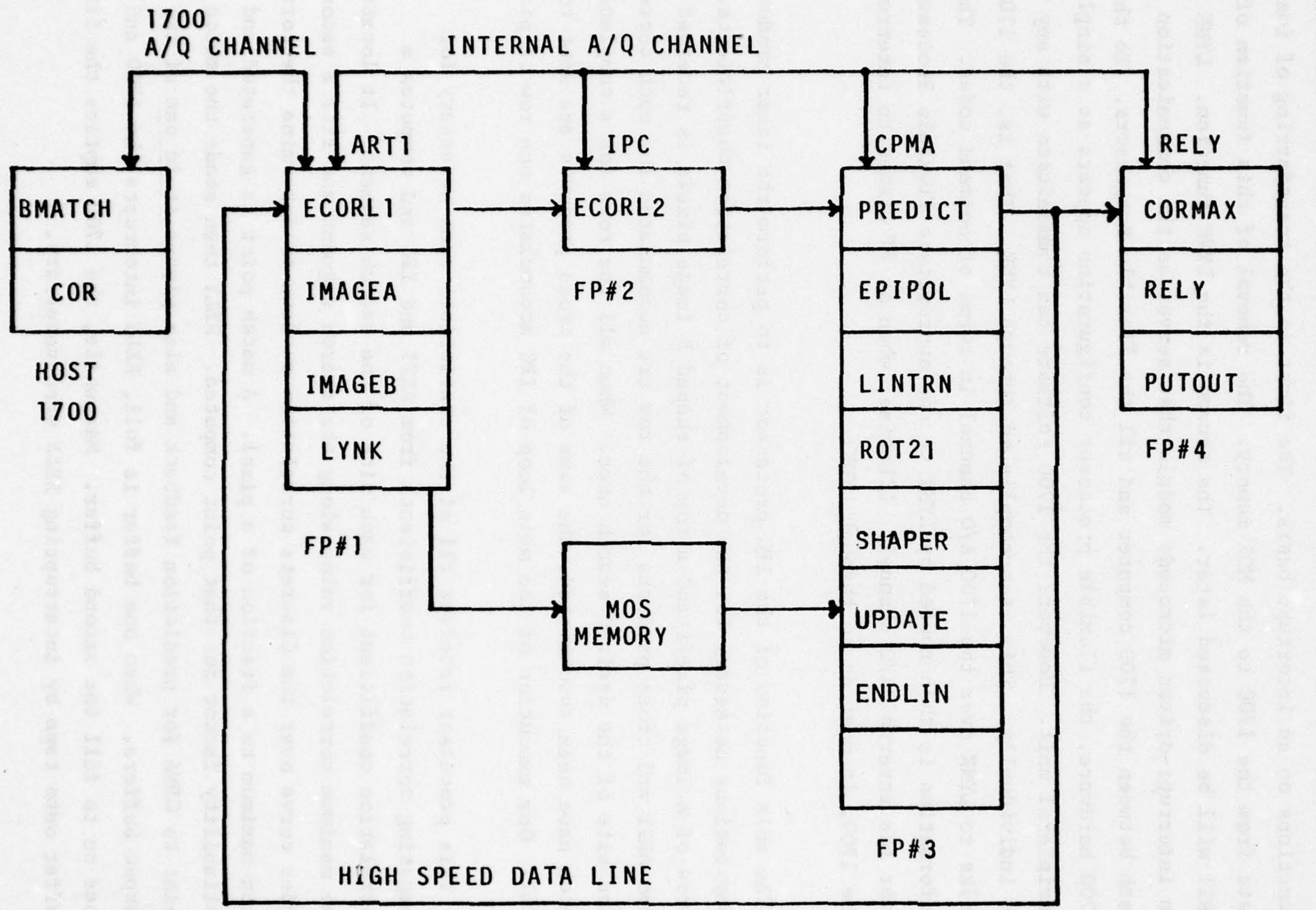


Figure 4-2 Logical Module Distribution

D3938

When not performing this correlation arithmetic, ART1 has two additional functions on an interrupt basis. The first is the transferring of image data from the 1700 to the MOS memory. The removal of this function of ART1 will be discussed later. The second is the LYNK function. LYNK is an interrupt-driven microcode module that serves as the communication path between the 1700 computer and all the Flexible Processors. To the 1700 hardware, the flexible processor configuration appears as a single peripheral unit. However, the 1700 software can communicate with any FP individually. This is accomplished through LYNK. That is, the 1700 talks to LYNK over the 1700 A/Q channel in terms of command codes. This information is then routed by LYNK to the appropriate Flexible Processor over the internal A/Q channel. Likewise, when an FP wishes to interrupt the 1700, it must do so through LYNK.

IPC - The sole function of the IPC processor is to perform the inner product computations necessary for the development of correlation coefficients. A row of A image pixels and a row of shaped B image pixels is received from ART1 and cross products for the row are accumulated for each correlation site of the desired search area. When all the rows for a complete patch have been accumulated, the sums of the cross products are sent to RELY. One execution of the main loop of IPC accumulates one row of pixels.

RELY - This processor receives all of the statistical sums necessary for computing correlation coefficients from ART1 and IPC and computes a correlation coefficient for each site of the search segment. It locates the maximum correlation value along the search segment and fits a second order curve over the discrete correlation surface to determine the correlation maximum to a fraction of a pixel. A match point is generated and the reliability factor for that point computed. RELY then sends the match point to CPMA for prediction feedback and also places it in one of two output buffers. When one buffer is full, RELY interrupts the 1700 and goes on to fill the second buffer. Meanwhile, the 1700 empties the first buffer onto tape by interrupting RELY when necessary.

HOST 1700 - The Control Data 1700 host computer is the primary link between the operator, the peripheral I/O units and the array of parallel processors. The COR operating system running on the 1700 performs the benchmark functions in two modes: An interactive operator command mode, and an interrupt-driven service mode. These two modes are outlined in Figures 4-3 and 4-4. In the command mode, the system, upon command, loads all flexible processors with their corresponding microprograms, clears and starts all FPs running, receives algorithm initializiation parameters coming in as part of commands, computes additional parameters from those given, initializes all FPs with their appropriate starting parameters, and starts the algorithm running. This initialization procedure is performed once per matching run. In the interrupt service mode, the 1700 fetches more image data, sends it to the MOS bulk memory through ART1, and also empties the match point buffers in RELY.

4.2 COMMUNICATION AMONG PROCESSORS

The backbone of the algorithm implementation on the Flexible Processor configuration is the nature of the communication between processors. Generally, the smaller the amount of communication overhead, the more efficient is the implementation. For this reason, efforts have been made to limit communication in the benchmark implementation to single-step flag settings or to transfers of just a few words of data. The exception to this, of course, is the transfer of image data from processor to processor.

A breakdown for all the possible communication paths in the implementation follows. For each path there is an active processor ("from" column) and a passive processor ("to" column). The communication description states the function of the active processor. Generally, when communication is initiated from an active processor to a passive processor, the passive processor does respond, but this response is typically a hardware logic function internal to the communication channel. Therefore, this response is not included in the breakdown because no overhead cost is incurred.

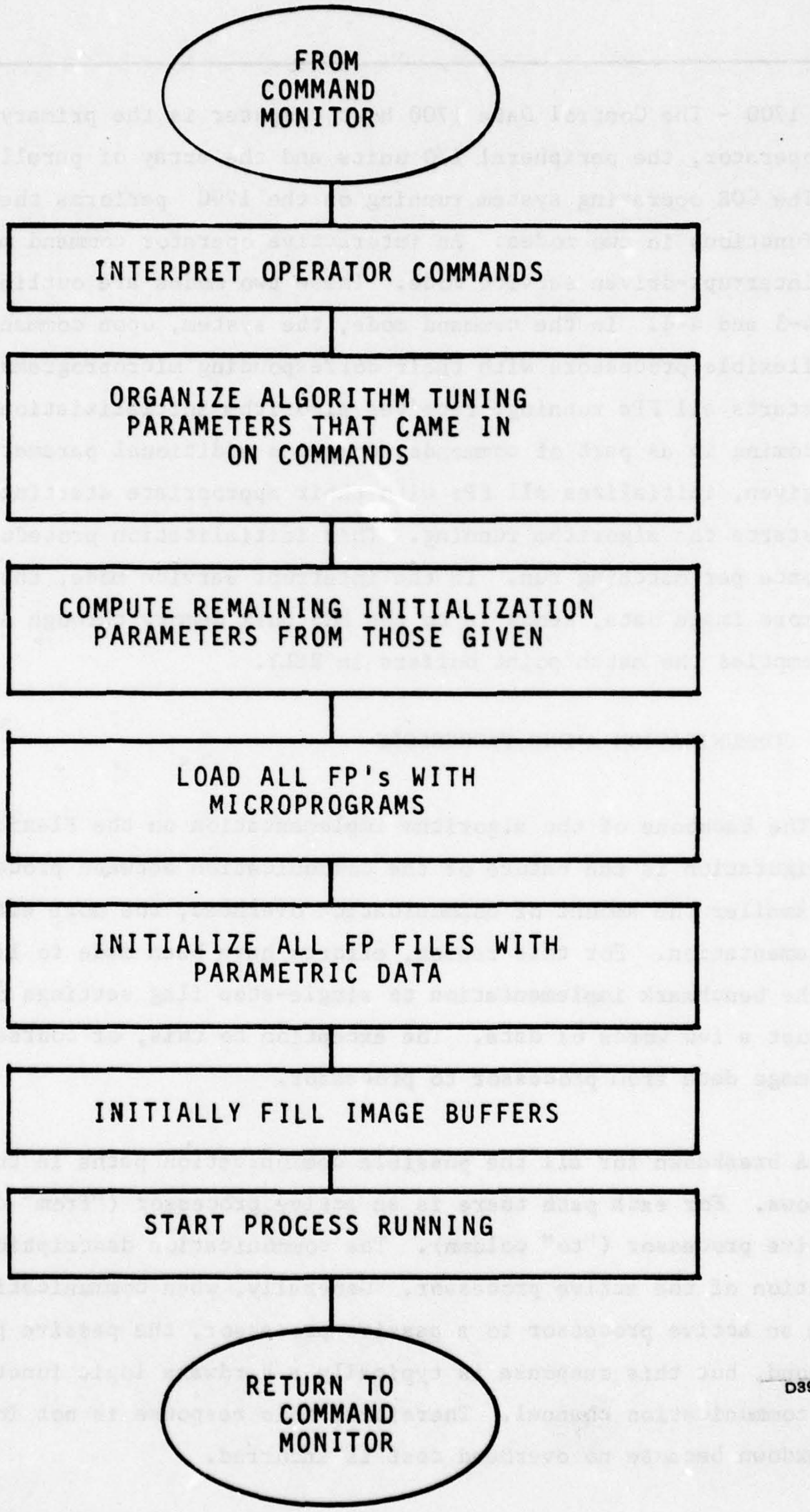


Figure 4-3. Host 1700 Command Mode

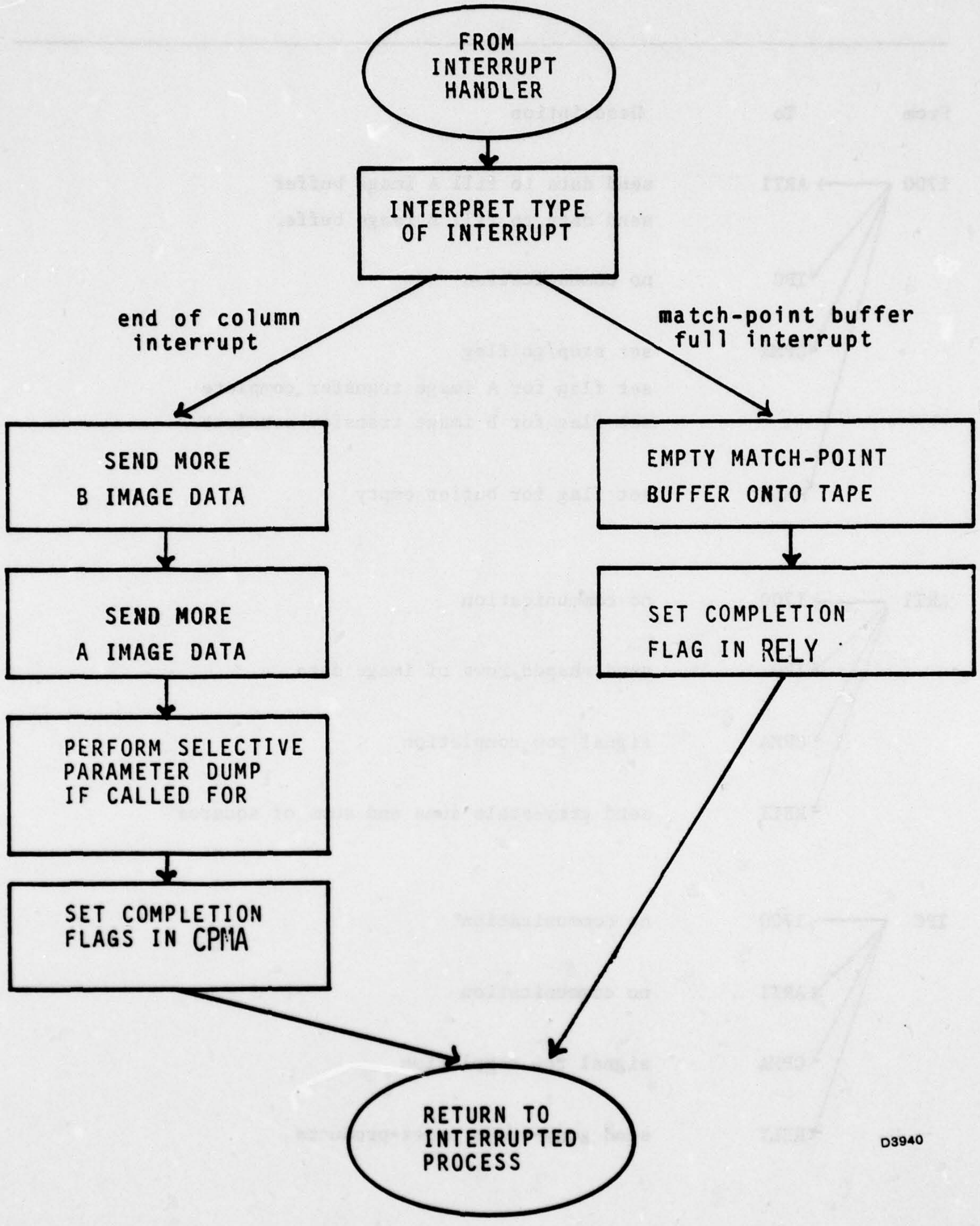


Figure 4-4. Host 1700 Interrupt Mode

From	To	Description
1700	ART1	send data to fill A image buffer
		send data to fill B image buffer
	IPC	no communication
	CPMA	set stop/go flag
		set flag for A image transfer complete
		set flag for B image transfer complete
	RELY	set flag for buffer empty
ART1	1700	no communication
	IPC	send shaped rows of image data
	CPMA	signal row completion
	RELY	send gray-scale sums and sums of squares
IPC	1700	no communication
	ART1	no communication
	CPMA	signal row completion
	RELY	send gray-scale cross-products

CPMA	→ 1700	send end of column interrupt
	→ ART1	send A image row and shaped B image row
	→ IPC	no communication
	→ RELY	send predicted matchpoint location
RELY	→ 1700	send match buffer full interrupt send match point data
	→ ART1	no communication
	→ IPC	no communication
	→ CPMA	send match point data for prediction update

4.3 OPERATING SYSTEM

The operating system chosen to run on the 1700 host computer to support the benchmark algorithm is called the COR Operating System. This operating system is designed primarily as a vehicle for communicating with and controlling Flexible Processor configurations. Some of the basic capabilities of the operating system are as follows:

- A command monitor for interpreting and executing operator commands that are entered via a CRT character device or a teletype.
- A file system for creating and editing microprograms and command files.
- A set of hardware diagnostic programs to check the integrity of Flexible Processor components.

• A set of utility programs to alter FP micromemory, dump FP storage files, and perform various other debugging operations.

A microcode assembler is also available, but it resides in the standard CDC 1700 MSOS Operating System.

The COR Operating System is organized in terms of a resident portion and a series of overlays. The command monitor, interrupt handlers, and FP utility programs make up the resident portion. The file editor and other file system utilities as well as the hardware diagnostic programs reside in overlays. User programs such as the benchmark control program, are also implemented as overlays.

Besides the standard set of commands that is interpretable by the command monitor, the capability exists for defining user commands that are applicable to the user's overlay. As an additional capability, commands may be grouped together in a command file; and by issuing the file name as a single command, the operating system is directed to interpret and execute each individual command in the file sequentially.

This command file capability is the vehicle for implementing various tuning parameter options for the benchmark stereo matching algorithm. A series of parametric commands have been defined for operator entry of algorithm tuning parameters, and a series of action commands defined to control the operation of the algorithm. Thus, for a given set of sensor images to be matched, the tuning parameters are determined and entered, on parametric commands, into a command file. Then, for all subsequent runs of that imagery, only the command file name is necessary to initiate the runs. This capability eliminates the need to enter parameters for each similar run.

Following is a list of the commands that have been implemented for the tuning and control of the benchmark algorithm. The basic command format is the command name followed by a sequence of integers or decimal fractions that is variable, depending on the command. All the benchmark and block-matching

commands have been prefixed with BM to distinguish them from the standard COR command repertoire.

<u>Parametric Commands</u>:

BMBASE	-	relative orientation baseline vector
BMFOC	-	focal length of cameras
BMORM	-	relative orientation matrix
BMFXA	-	A image interior orientation transformations
BMFXB	-	B image interior orientation transformations
BMABUF	-	A image buffer size and characteristics
BMBBUF	-	B image buffer size and characteristics
BMGRID	-	matching grid specification
BMPCH	-	correlation patch size and search specification
BMPWGT	-	prediction weight functions
BMRELT	-	reliability cutoff values
BMKPN	-	number of initial match points for startup
BMMTCH	-	initial match points to start process

<u>Action Commands</u>:

BMLDA	-	load all FP's with microprograms
BMSTUP	-	set up initialization parameters for FP's
BMINIT	-	initialize all FP's with startup parameters
BMGO	-	start algorithm running
BMHALT	-	stop algorithm
BMRESM	-	restart algorithm after BMHALT
BMDFLT	-	set up parameters for A vs A autocorrelation
BMDMP	-	dump selected matching parameters

5.0 OPERATIONAL FLOW

To further explain the internal operation of the benchmark algorithm and system, the following sections outline the basic sequence of processing events. These events occur in two separate environments; the user/host computer environment and the parallel array environment.

5.1 USER/HOST COMPUTER ENVIRONMENT

The interactive involvement of the user and the host 1700 in the benchmark algorithm is outlined in the following steps. All of these steps occur in a matter of seconds, or as fast as the user can respond at a keyboard.

1) The user loads the resident monitor of the COR operating system from disk. The system maintains in files on the disk all microcode binaries, all command service programs, and the major overlay to run the benchmark matching algorithm.

2) Via a CRT character display, the user loads the four Flexible Processors with their microprograms and starts them running in their idle loops. This is performed by the command BMLDA.

3) The user then proceeds to execute the command file that contains the algorithm tuning parameters in the form of parametric commands. The user can also enter single parametric commands to alter the parameters entered from the command file. For example,

   BMGRID 32 144 8 16 166 10

   defines the matching grid over which to correlate; that is, find match points from image scan line 32 to line 144 in steps of 8 lines and from pixel 16 to pixel 166 in steps of 10 pixels. As these parametric commands are executed, the parameters are reformatted and placed in a storage block in 1700 memory.

4) When all parametric commands have been processed, the user issues

the BMGO command. This command performs the following functions:

- The basic tuning parameters entered above are used to compute the matching initialization values which are reformatted to be compatible with the FP's and are organized into blocks in the 1700 memory. These blocks are images of the FP large file registers. The reformatting is necessary to achieve various double precision number scalings in the FP's and because of the fact that the 1700 is a one's complement machine while the FP is a two's complement machine.

- The blocks of initialization values are transferred to the FP large files.

- The image buffers in the MOS bulk memory are initially filled.

- The 1700 interrupt system is enabled.

- The GO flag is sent to CPMA to start the block matching algorithm running.

- Control returns to the COR command monitor and the 1700 waits for interrupts.

5) The algorithm terminates when all match points over the selected grid have been generated.

6) While the algorithm is running the operator can suspend execution by issuing the BMHALT command and can continue by issuing the BMRESM command.

7) The algorithm can also be placed in a step mode by the BMDMP command such that the algorithm stops after every column of match points and a decimal dump of user-selected matching values appears

on the CRT. This allows the user to monitor the internal environment of the parallel matching process.

5.2 PARALLEL ARRAY ENVIRONMENT

In this environment CPMA functions as the executive processor of the parallel array. CPMA maintains all of the prediction parameters and all of the recent correlation history in its large files. The other processors are more or less controlled by CPMA in performing tasks to generate data that is used by CPMA to keep the process running. Following is the sequence of events that occur in the four Flexible Processors. The process names, like CPMA, will be used to describe the action of the processors. It must be kept in mind that most of these events occur simultaneously in different FP's.

1) At the start of a column of match points, CPMA first determines whether the data is available in the image buffers for that column. If it is available, CPMA then sends an interrupt to the host 1700 to start filling the buffers for the next column.

2) For each correlation patch of the column, CPMA fetches the prediction data for that patch from its files and sends it to RELY. The same data is then used to compute all the shaping parameters for B image data. CPMA then accesses from MOS bulk memory one patch row of A image data and the corresponding shaped row of B image data. These rows are sent over the high-speed data line to ART1. At this point CPMA goes on to access and shape the next row of data.

3) When ART1 receives the A row and shaped B row of image data, it immediately passes the same data to IPC over the high-speed data line. The ART1 proceeds to accumulate the gray-scale values of the row of pixels into registers that correspond to an A patch sum, an A patch sum of squares, B patch partial sums, and B patch

partial sums of squares. When a row is complete, ART1 signals its readiness to CPMA. When all the rows of a patch are complete, ART1 sends the resulting sums to RELY over the internal A/Q channel. While all of the above is occurring, a background process in ART1 is using the LYNK module in an interrupt mode to relay the image data for the next column from the host 1700 to MOS bulk memory.

4) When IPC receives the A row and shaped B row of image data from ART1, it starts accumulating gray-scale cross products in its registers. More than one sum of cross products is required because correlation coefficients will be formed for a number of sites on each side of the predicted site along the epipolar line. Therefore, every A image pixel in the row must be multiplied by several corresponding B image pixels. The number of multiplies here is dependent on the number of correlation sites. When a row is complete, IPC signals its readiness to CPMA. When all the rows of a patch are complete, IPC sends the sums of cross products to RELY over the internal A/Q channel.

5) When RELY receives the statistical data from ART1 and IPC, it forms a correlation coefficient for each site of the search segment surrounding the predicted site. Then RELY finds the maximum value of these correlation coefficients and fits a parabolic curve over this maximum value and the values at the two adjacent sites. In this way the correlation maximum and therefore position of best match is determined to a fraction of a pixel. RELY next computes the reliability of the match based on the preset reliability cutoff values and the statistics from the match. A correction to the match point is made if necessary. Then a final match point 5-tuple (x, y, u, v, r) is formed and deposited in a match point buffer. The u portion is sent to CPMA over the internal A/Q channel for future predictions. When one match point buffer becomes full, RELY sends an interrupt to the host 1700 to empty the buffer and subsequent match points are placed in a second buffer.

The above events comprise the majority of processing activity of the parallel array. If the algorithm were stopped arbitrarily at row k of patch j of column i, the following would be observed:

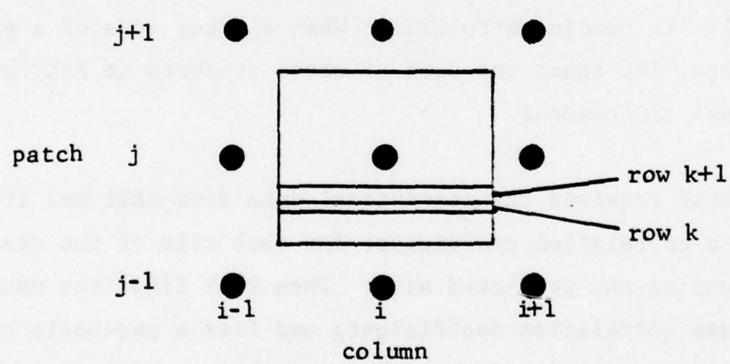
- The host 1700 is transferring image data for column i+1 into MOS bulk memory.

- CPMA is accessing and shaping row k+1 of patch j.

- ART1 is accumulating sums for row k.

- IPC is accumulating cross products for row k.

- RELY is developing a match point for patch j-1.



Additional processing events are as follows:

6) After the first patch of a column, that is, when the match point for the last patch of the preceeding column has been completed by RELY, CPMA updates its correlation history and prediction mechanism based on the entire column of new-found match points.

7) In addition, CPMA corrects any patches that may have wandered in the previous column. This is done by comparing a patch's position with the average of its neighbors' and correcting those patches whose position deviates by more than parameter "wandering block tolerance".

8) Between the patches of a column, CPMA uses relative orientation of the exposures, the interior orientation transformations, epipolar geometry, and previous matching history to define the predicted v coordinate of the next patch.

6.0 BENCHMARK TIMING

The purpose of this section is to specify the timing of each microprogram and the overall throughput rates of the benchmark algorithm. In general, the algorithm timing and throughput rates are variable, depending on certain tuning parameters such as correlation patch size and match point interval size. Below is a list of the timing variables that appear in the timing formulas.

HXA - integral half patch size in the line direction (X)

HYA - integral half patch size in the pixel direction (Y)

SX - integral half of the number of correlation search sites along the epipolar line.

For example, for a correlation patch that is 21 lines by 11 pixels, HXA = 10 and HYA = 5. If correlation is to occur at seven sites, including a predicted site and three sites on each side, SX = 3.

NSPX - number of grid intervals in the line direction (X) that fall within a single patch. Also, the number of subpatches in X that have different shaping parameters.

NSPY - number of grid intervals in the pixel direction (Y) that fall within a single patch. Also, the number of subpatches in Y that have different shaping parameters. The total number of subpatches in a patch, then, is NSPX*NSPY.

KPN - number of patches or match points per column.

Figure 6-1 illustrates the dimensionality of these variables.

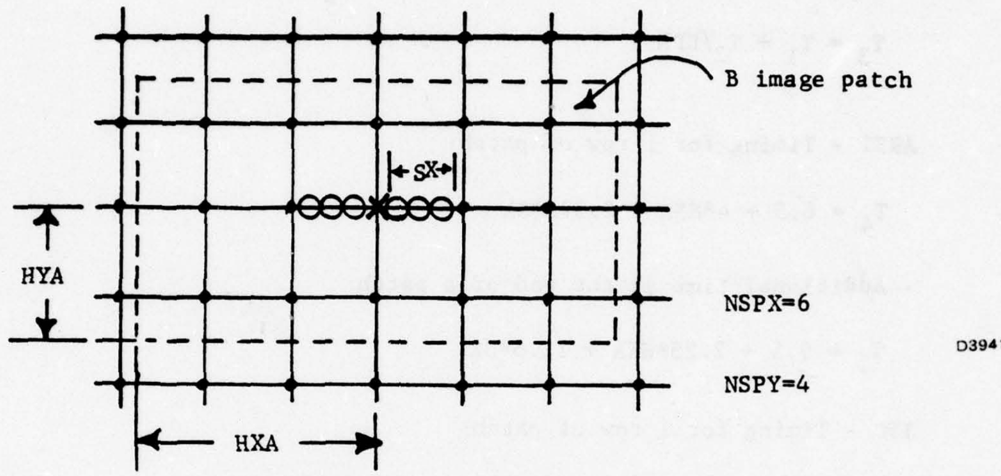


Figure 6-1. Timing Variable Definition

The following is the timing breakdown for each processor in the parallel array. All the times are expressed in microseconds.

CPMA - Timing for one patch:

$$T_1 = 204.88 + 3.313*NSPX + 4.625*NSPY + 2*NSPX + NSPY$$
$$+ (HYA + 1) \quad [25.375 + 23.375*HXA + 19.75*SX + 3.5*NSPX]$$

- Additional time at the end of a column:

T_2 = 18.25 + 5.438*NSPY + 15.75*KPN

- Total time per patch including end of column:

$T_3 = T_1 + T_2$/KPN

ART1 - Timing for 1 row of patch:

T_4 = 6.5 + 4*HXA + 2.375*SX

- Additional time at the end of a patch:

T_5 = 9.5 + 2.25*HXA + 11.5*SX

IPC - Timing for 1 row of patch:

T_6 = 2.5 + 1.625*HXA + 1.75*SX + 1.75*HXA*SX

- Additional time at the end of a patch:

T_7 = 2.5 + 4*SX

RELY - Timing for 1 patch:

T_8 = 110.125 + 106.5*SX

All of these times were determined by accumulating the number of instructions in each microcode module and multiplying by the Flexible Processor instruction cycle time, .125 microseconds.

In order to determine the time per patch of the entire benchmark, the parallel interleaving of the times for individual processors must be considered. CPMA is the most heavily loaded processor and thus runs the longest per patch. The critical time for this processor is the time required to access a row of A image data and to access and shape a row of B image data. This

time for a single row pair is:

$$T_{1A} = \left[25.375 + 23.375*HXA + 19.75*SX + 3.5*NSPX\right] /2$$

Now, the ART1 and IPX row times, T_4 and T_6 respectively, are interleaved with T_{1A}. Likewise, the ART1 and IPX end of patch times T_5 and T_7, are interleaved with the remainder of T_1. The total patch time for RELY, T_8, is also interleaved with T_1. Therefore, the total time per patch for the benchmark is determined solely by CPMA. This is time T_3.

When the benchmark was completed and analyzed, results showed that the majority of algorithm running time was spent accessing and shaping rows of image data. This was somewhat surprising. In Phase B of the program it was determined that the majority of time would be spent by the processor responsible for accumulating the statistical sums for correlation. Therefore, this task was distributed between ART1 and IPC to achieve a higher degree of parallelism and thus reduce the time spent in this task. This distribution caused the shaping portion of CPMA to emerge as the major time consumer.

In order to analyze the benchmark timing in more practical terms, an individual case of algorithm tuning will be considered. In processing the 1:48000 Phoenix stereo model, it was found that a 21 x 21 pixel patch over an eight-line by ten-pixel matching interval provided adequate algorithm tuning for collecting digital terrain data. This represents a rather conservative case, because the algorithm runs much faster with smaller patches. For this case the digital scan interval is 25 micrometers and the stereo overlap area is approximately nine inches by five inches. Therefore, a pair of images that are 9000 by 5000 pixels must be matched. At the specified grid interval, this involves the generation of approximately 562,500 match points. The pertinent tuning considerations for this case are summarized in Table 6-1.

6.1 INPUT/OUTPUT CONSIDERATIONS

In the preceding discussion, all of the timings refer to parallel processor times alone. The overhead time necessary to transfer image data to and from the parallel processing array and the system peripheral units was not considered. A number of options exist for this image data input; the benchmark system contains four CDC 608 tape drives, two 854 disk drives, and two storage module drives (SMD). The SMD is by far the superior I/O device having an unformatted storage capacity of 80 megabytes and the capability for transferring one 16-bit word or two 8-bit pixels in 2.7 microseconds. In the original Phase B plan for the benchmark the input of image data into the system was to be from a SMD to the Line Buffer Memory, which is shown in Figure 3-2. All the Flexible Processors can easily access this memory, and there is little host 1700 intervention in the data transfer process. However, it was learned that the SMD's, their controller, and interface would not be available in time for the benchmark. Therefore, the input implementation for the benchmark is from tape as an interim measure. At the time of this writing the SMD's are part of the system and functioning properly. Eventually, the tape input will be removed from the benchmark algorithm and the data accessing modules will be converted to receive input from the SMD's through the line buffer memory.

Currently, input image data enters the algorithm from tape at a rate of one 16-bit word or two 8-bit pixels every 533 microseconds. At this rate the data for the next column of correlation patches can be completely transferred in the time it takes the algorithm to process 1092 patches of the current column under the tuning conditions for the above sample case. In most of the test cases run to date, in which the number of patches per column was less than 1092, the algorithm ran faster than the tapes could move; the Flexible Processors spent most of their time in idle loops waiting for more data.

Under 854 disk implementation, image input occurs at the rate of one 16-bit word every 16 microseconds. At this rate, a column of image data can be transferred in the time required to process 33 patches. Using the

SMD, however, the transfer rate is one word per 2.7 microseconds, and a column can be transferred during the processing of just six correlation patches. The net result of this I/O analysis is that the benchmark algorithm is not I/O bound when using the SMD as an I/O source. Thus the parallel array spends no extra time waiting for data, and time T_3 above is an accurate prediction of the total benchmark throughput.

TABLE 6-1. TIMING SUMMARY FOR A REPRESENTATIVE MAXIMUM CASE

Patch size	2 x 21 pixels
Matching grid interval	8 x 10 pixels
Correlation search size	5 sites
Image size	5000 x 9000 pixels
Match points per model	562,500
Benchmark time per match point	.0037 seconds
Benchmark time per model	35 minutes
CDC 6400 time per model	20 hours
Speed increase of benchmark CDC 6400	34 times
CDC 6600 time per model	11 hours
Speed increase of benchmark over CDC 6600	19 times
Benchmark throughput rates	270 match points/ second 119,000 pixels/second
Equivalent add operations per patch	29,600
Equivalent add operations per pixel	67

7.0 LEVELS OF PARALLELISM

Three levels of parallelism are generally considered when dealing with the implementation of algorithms on parallel processing hardware, particularly on the Flexible Processor. These are the instruction logic level, the processor level, and the systems level.

On the instruction logic level, parallelism arises from the fact that more than one operation can be performed in a single microinstruction. The Flexible Processor is organized such that there are two separate data buses that connect two input files, two temporary files, two large files, and two adders. Therefore, 16-bit operands may be routed among these components independently and in parallel over either bus. For example, a 16-bit number in a temporary file location may be sent to the adder over one bus; while, at the same time, two 8-bit pixels may be transferred from an input file location to a large file location on the other bus; while, at the same time, the hardware jump stack may be incremented or decremented. All of this occurs in one instuction cycle, the sequence of events being synchronous with the internal timing scheme of the instruction. Some of this parallelism is lost, however, when 32-bit operands are processed because both buses are required.

It is at this level that a microprogrammer can exploit all the devices of his art. To implement a fast algorithm, it is of the utmost importance to sequence the individual operations of the algorithm in such a way that the maximum parallelism is achieved at the instruction level. A truly optimized algorithm is one in which both buses are kept busy through most of the instruction cycles.

On the processor level of parallelism, how much parallelism can be achieved among the processor in a configuration is the main concern. In terms of the benchmark, much of this level has been discussed in previous sections. One additional point is that the stereo matching algorithm has been implemented on the benchmark configuration as a parallel, asynchronous

process. In this way, each Flexible Processor performs its given task in the algorithm individually and simultaneously with the other processors; when it is finished, it waits on a flag or interrupt for more data to perform its task again.

Previous Flexible Processor systems have been designed to operate in a synchronous mode. That is, the microprograms have been written with the exact number of instructions in each processor such that all processors are synchronized by the inherent instruction timing. In this way, when one processor is ready to send data to another processor, this second processor is always ready to receive it. Clearly, in this synchronous approach there is no need for the overhead required to perform handshaking between processors; one processor need not query a second processor as to his readiness and the second processor doesn't have to spend time to reply that it is ready or still busy, as in the asynchronous approach.

Thus, the synchronous approach can conceivably run faster than the asynchronous approach, but there is much more flexibility inherent in the asynchronous approach. Here, additions to the microprograms or changes to the basic tuning parameters, such as increases in patch sizes which require longer running loops, do not upset the timing of the entire algorithm. Thus, there is certain modularity in this type of parallelism; one module can be altered without significantly affecting the other modules.

The third level of parallelism, the systems level, exists if Flexible Processors are configured into separate, but duplicate, processing channels. Here, the microprograms are the same for each channel but the algorithm is run in parallel over different sections of digital image data. For example, if there are four processing channels available, then the imagery can be partitioned into four adjacent sections such that one channel processes one section simultaneously with the other channels and sections. Theoretically, then, a four channel system can realize a fourfold increase in total throughput rate over a one channel system. However, additional overhead is incurred if it is necessary to link the parallel channels together

with parametric data. Due to the limited number of Flexible Processors available for the benchmark, the block-matching algorithm has been implemented in a one channel configuration.

7.1 IMPACT ON ALGORITHM DEVELOPMENT

There are some basic operational differences between the parallel implementation of the stereo matching algorithm on the Flexible Processors and the general-purpose implementation of the algorithm on the CDC 6400. Most of these differences have arisen from the fact that a basically sequential algorithm has been converted for parallel implementation.

To illustrate this, consider Figure 7-1 which shows the order of patch processing in the sequential algorithm. For the case illustrated, there are four patches per column to be matched. The number in each patch denotes the order in which the patch is completely processed in relation to the other patches. In other words, in the sequential algorithm, which was the predecessor of the benchmark parallel algorithm, patch j of column i is processed completely before moving on to patch j+1, j+2, j+3, and so on. This is the natural sequence of processing for a sequential machine such as the CDC 6400 and for a sequential language such as Fortran.

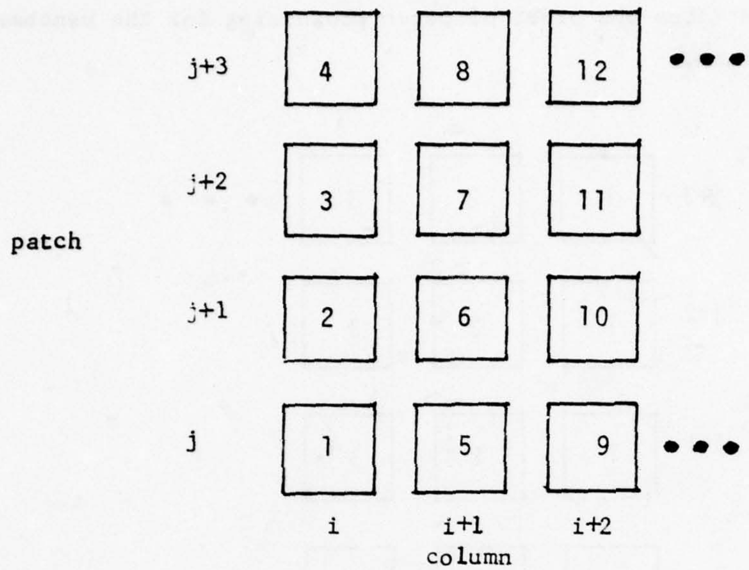


Figure 7-1. Sequential Implementation

Now, as a result of the sequential nature of the algorithm, certain correlation strategies emerge. That is, the correlation of any patch can be facilitated by the algorithm's knowledge of what occurred during the correlation of the previous patches in the sequence. In particular, the parameters for shaping the correlation patch j+1 for column 1+1, number six in sequence, are predicted from the already correlated patches j+1, i and j, i+1, numbers two and five in sequence, respectively. These are the nearest patch neighbors and provide the most valid predictions, particularly when the patches overlap.

If this sequential strategy was transferred directly to hardware capable of parallel processing, such as the benchmark configuration, the increase in speed and throughput over the sequential implementation would not be great. Parallelism would occur primarily on the first level, the instruction logic level. Processors would be assigned their individual tasks, but now would perform the tasks in sequence rather than simultaneously with the other processors.

Figure 7-2 illustrates the order of patch processing for the benchmark parallel implementation.

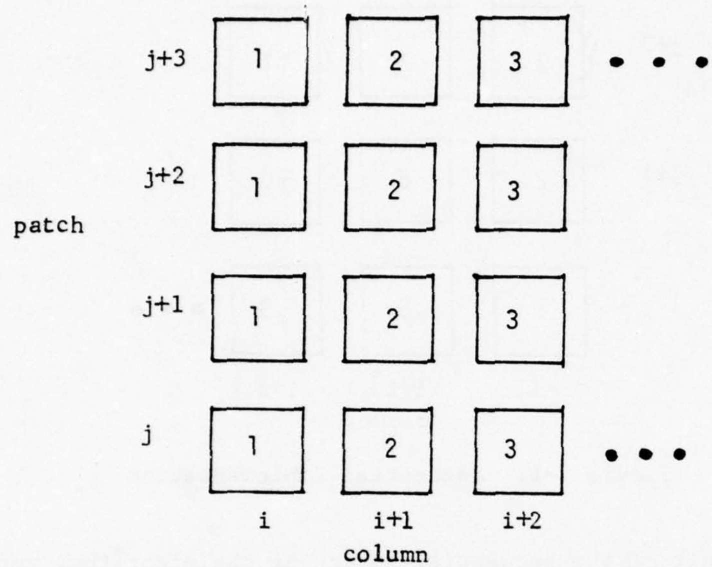


Figure 7-2. Benchmark Parallel Implementation

Here it is not possible to use patch j, i+1 in predicting parameters for patch j+1, i+1. All the patches of a column can be considered as being predicted and correlated simultaneously. Therefore, valid predictions can only be derived from the preceding column. In this case, the nearest neighbors to patch j+1, i+1 are, first, patch j+1, i, then also patches j, i and j+1, i.

Thus, a tradeoff exists in this particular algorithm case involving correlation strategy and parallel implementation. The effects of the strategy change have proven to be rather image-dependent. That is, the parallel strategy is less effective than the sequential strategy in certain hard-to-correlate areas of stereo scenes. But these areas can generally be handled by adjustments to the basic algorithm tuning parameters.

But there is a hypothetical alternative to the basic tradeoff that combines the strategy of the sequential implementation with the advantages of the parallel implementation. The order of patch processing for this alternative is illustrated in Figure 7-3.

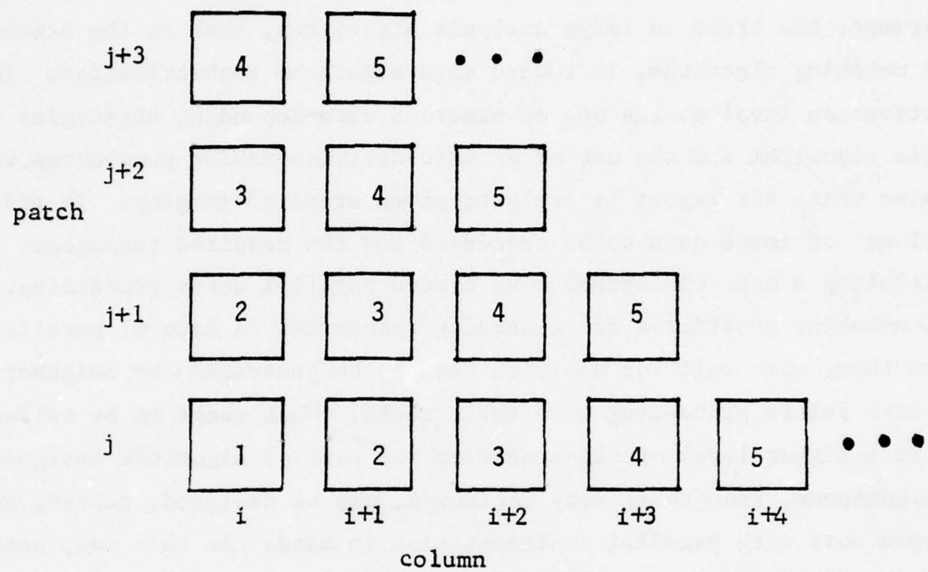


Figure 7-3. Hypothetical Parallel Implementation

Instead of a column-oriented parallelism, there arises a diagonal parallelism that spans across many columns and many scan lines of image data. One can see that as the number of patches per column increases, so does the complexity of the algorithm overhead and the data structures needed to achieve the parallelism. So, while the basic tradeoff involving correlation strategy and parallel implementation has been solved, new and possibly more serious tradeoffs emerge involving overhead complexity and time, number of parallel processors, and the size of local and bulk memory space. There are also many ramifications regarding internal algorithm design.

7.2 CONCLUSION

The above discussion has been presented as a particular algorithm example of the more general class of considerations that are involved in the design and implementation of parallel image processing algorithms. For future development, the trend in image analysis algorithms, such as the benchmark stereo matching algorithm, is toward more algorithm sophistication. This sophistication involves the use of numerous data-dependent strategies within a single algorithm and the use of dynamic decision-making procedures that determine which strategies to apply to given areas of imagery. In addition, the volume of image data to be processed and the required throughput rates are dictating a more widespread move toward parallel array processing. Yet, decision-making procedures are generally sequential in nature; parallel processors then, must wait for decision data to be generated by neighboring processors before proceeding with their tasks. What seems to be called for then, is a higher level of cleverness on the part of algorithm designers so that algorithms, from their very beginning, may be designed, tested, and developed more with parallel implementation in mind. In this way, better solutions will emerge for handling the complex tradeoffs that are part of parallel processing technology.

Digital Cartographic Study and Benchmark
Fourth Interim Technical Report

Prepared For:

U.S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia

Contract DAAG53-75-C-0195

Prepared By:

D.J. Panton

July 1977

Digital Image Systems Division
Control Data Corporation
2800 East Old Shakopee Road
Minneapolis, Minnesota 55420

TABLE OF CONTENTS

LIST OF FIGURES

1.0 INTRODUCTION

This report is the fourth in a series of Interim Technical Reports that cover the development and implementation of a stereo matching algorithm that can be used in automatic terrain data collection. In particular, the results of Phase D of the contract are contained herein. The primary purpose of this Phase was to generalize the algorithm that was developed under Phases A, B, and C to handle more uncontrolled cases of central perspective photography and to lay the groundwork for handling non-central perspective photography.

Previous developments and algorithm logic modifications have been reported in a rather piecemeal fashion over the first three Phases. This report combines all these developments into a consistent description of the matching algorithm as it appears to date, including the modifications of Phase D.

Throughout the continued development, the matching algorithm has become somewhat of a terrain and sensor analysis tool rather than a strict stereo compilation technique. Under the tuning parameter concept, it is possible to apply the algorithm to a wide range of sensor, image, and terrain conditions. Using the algorithm's built-in reliability analysis, it is possible to assess the difficulty and quality of automatic matching under these varied conditions. It has been found that no single digital technique can approximate the capability of a human stereo compiler when faced with varied sensor records and diverse image and terrain events contained in these records. This is the justification for algorithm tuning. Moreover, the stereo conditions that are optimal for human compilation are not necessarily optimal for automatic matching and vice versa. This is the rationale behind reliability monitoring. The overall objective is to exercise the algorithm under representative experimental conditions so that decisions can be made and parameters acquired regarding what actually is optimum for automatic stereo mapping.

1.1 Underlying Assumptions

The basic philosophy behind the block matching system design is that the ideal situation for matching the images of a stereo pair is to match each pixel of one image individually with its corresponding position on the other image. However, as long as the correlation coefficient is used as the similarity metric between the images, matching one pixel with one pixel is not possible because of the low statistical significance of such a small sample. Therefore, it is necessary to measure the similarity of a group of pixels surrounding the pixel in question. The size and shape of this group, or image patch, must be chosen carefully and may vary from image to image and also from area to area in the same image. The underlying objective is to choose as small a patch as possible such that the local image noise and lack of feature content do not dominate the value of the correlation coefficient for that patch.

The correlation and consequent match point determination of an individual patch is not independent of neighboring patches and match points.

The correlation of image areas introduces a certain averaging effect on the actual point that is matched; the effect increasing as the size of the areas increases. Therefore, it is necessary to shape the image data within a correlation patch so that it conforms to all the match points in the vicinity. In this way, the averaging effect is confined to small linear segments between the match points. In addition, the correlation patch is shaped such that its projection in three-dimensional space approximates the terrain shape as closely as possible.

In matching images, heavy reliance on values of the correlation coefficient alone is a rather unreliable approach in terms of the accuracy of any given match point. Driven solely by correlation maximum searching over extended areas, patches and match points can wander considerably from their true positions depending on the geometric distribution of image noise and high frequency signal components. Therefore, it is necessary to apply a great

deal of geometric constraint to correlation patches and match points based on known geometric parameters and the continuity and slope limits of natural terrain. Stereo frame imagery offers an unique opportunity in this respect because of its inherent, well-defined geometry.

The concept of a coarse correlation search followed by a fine search is less than desirable in most cases because of the aforementioned unreliability of the correlation coefficient alone. The better approach seems to be to perform a fine correlation search followed by a refinement process. In general, well behaved image areas and terrain will match up well on the first fine correlation, and the results will be the same on subsequent refinements. Refinement is necessary only in difficult areas.

When matching stereo imagery, there are essentially six coordinate systems to consider. Depending on the circumstances of the stereo exposures, some of these systems may coincide with others; but for the general case, they must all be handled independently. These coordinate systems are:

- The digital scan coordinate system on the left image (denoted image A), two-dimensional.
- The photo coordinate system on image A defined by fiducial or other calibration marks, three-dimensional.
- The digital scan coordinate system on the right image (denoted image B), two-dimensional.
- The photo coordinate system on image B, three-dimensional.
- The epipolar coordinate system that relates image B to image A, two-dimensional.
- The model coordinate system, three-dimensional.

In stereo matching, it is desirable to define an evenly spaced grid in one of these coordinate systems to drive the matching process. The location of this grid is a factor in determining the number, complexity, and speed of the image processing functions that are collectively called stereo matching.

For example, pixel gray scale correlation must occur in a digital scan coordinate system so an evenly spaced grid in this system facilitates the resampling of image data. It has been found that this resampling and patch shaping are the most time-consuming operations in the matching process. However, prediction mechanisms that make use of epipolar geometry and other photogrammetric transformations deal with photo coordinate systems. Thus, an evenly spaced grid in one of these systems minimizes the prediction inaccuracy. In addition, match points from the stereo pair are intersected to produce terrain data in a model coordinate system. An evenly spaced grid defined here eliminates the need for postprocessing the terrain data. These concepts are summarized in Table 1-1.

In the stereo matching system that is described in this report, the evenly spaced grid is defined in the image A digital scan coordinate system. This has been done to minimize the resampling time and inaccuracy. However, the resulting terrain data must be postprocessed to produce a regularly spaced set of profiles in model space.

TABLE 1-1. PROCESSING REQUIREMENTS BASED ON THE PLACEMENT OF THE EVENLY SPACED GRID

Coordinate System Containing Evenly Spaced Grid	A Image Resampling and Shaping	B Image Resampling and Shaping	A Image Search	B Image Search	Post-Processing Terrain Data	Prediction Iteration
A Digital		Yes		Yes	Yes	
A Photo	Yes	Yes		Yes	Yes	
B Digital	Yes		Yes		Yes	
B Photo	Yes	Yes	Yes		Yes	
Epipolar	Yes	Yes		Yes	Yes	
Model	Yes	Yes	Yes	Yes		Yes

2.0 BLOCK MATCHING CONCEPTUALIZATION

The basic idea behind the current implementation of the matching method is to define an evenly spaced grid of points on image A, and then for each of these points to find its conjugate point on image B. The conjugate point is found by correlating a group of pixels surrounding the point on image A with a sequence of groups of pixels on image B. The groups of pixels are termed blocks or patches, and the sequence of patches on image B defines a correlation search area. These concepts are illustrated in Figure 2-1.

Digital scan lines are generally oriented normal to the flight direction and direction of major parallax. Processing occurs from left to right, or in the direction of increasing x coordinate on image A. A column is defined as a line of patches which lie on the same digital scan line or whose centers all have the same x coordinate on image A. The sequence of processing is to correlate all the patches of a column before moving on to the next column. This scheme allows a rather straightforward management of image A data. That is, the image A buffer window need only be wide enough to contain one correlation patch width of scan lines.

As can be seen in Figure 2-1, the conjugate line on image B of a single scan line on image A can be rather non-linear due to the effects of terrain relief. This conjugate line cuts across many digital scan lines on image B; thus, the image B buffer window must be considerably wider than the image A buffer window. In addition, the size, shape, and orientation of the conjugate patches on image B can be quite different from the nominally rectangular patches on image A, also due to terrain relief and the geometry of the stereo exposures.

The conceptual steps in finding a match point are as follows:

- Determine the next point to be matched along the evenly spaced grid of image A.

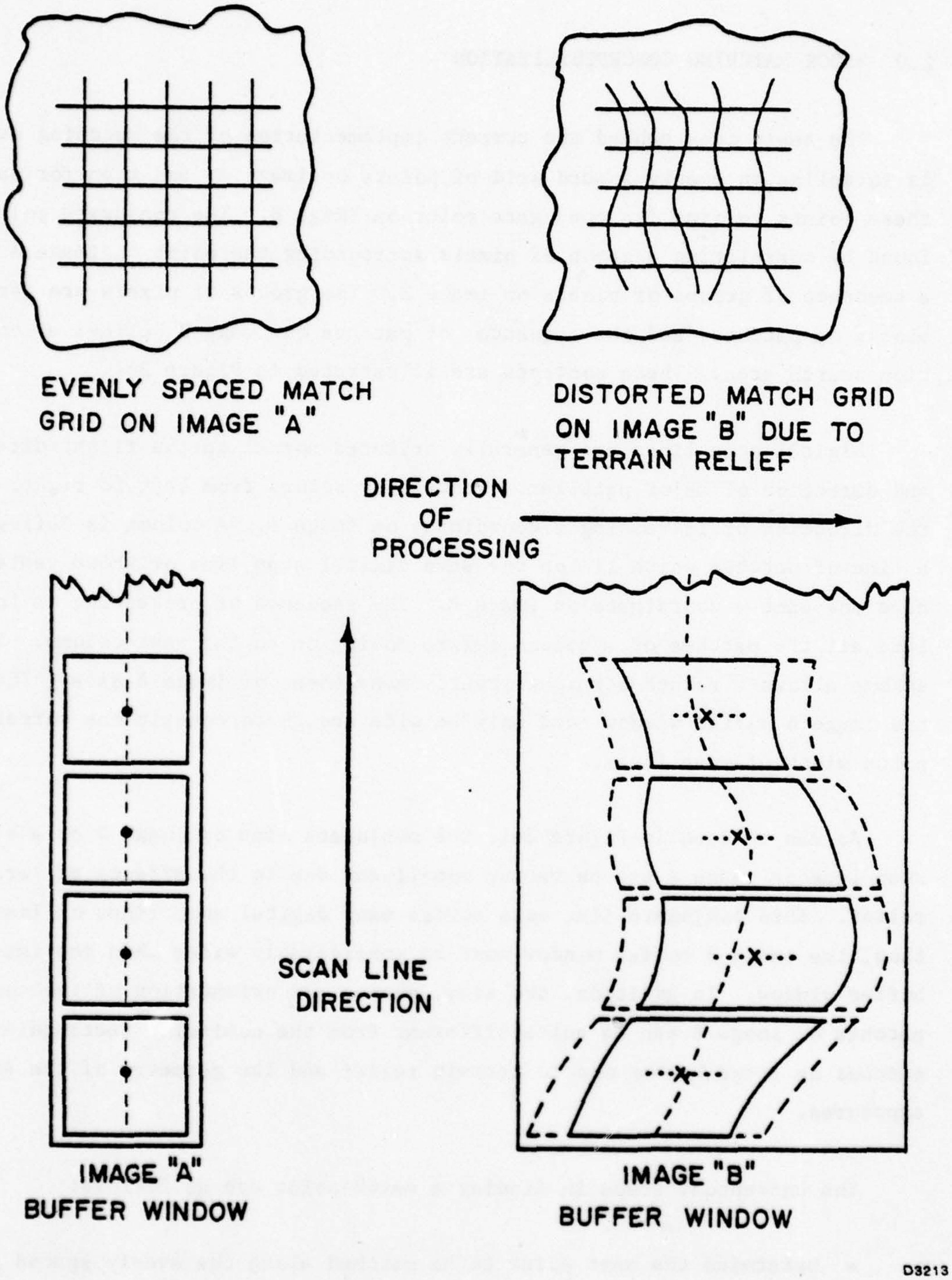


Figure 2-1. Block Matching Conceptualization

- Using epipolar geometry, compute the equation of the epipolar line passing through the point to be matched on image A.
- Compute the equation of the conjugate epipolar line on image B.
- Predict the location of the conjugate match point position on image B along the epipolar line using neighboring, previously matched points.
- Define correlation sites on each side of the predicted location on the epipolar line.
- Shape the image B patch and search area using previous and predicted match point information so that the image B patch most closely lies on the terrain and most closely conforms to the information content of the image A patch.
- Compute a correlation coefficient for the predicted match point location and for each search site.
- Determine the site of maximum correlation and fit a smooth quadratic function through it and its two neighbor sites to determine the correlation function maximum to a fraction of a pixel.
- Compute the reliability factor of the match point based on a set of reliability criteria.
- Apply a correction to the match point if it is excessively unreliable.
- Update the correlation history data and prediction mechanism based on this new match point and reliability.

The primary output of the matching algorithm is a file of five-tuples, one corresponding to each match point found. This five-tuple is illustrated in Figure 2-2 along with the basic terminology of the algorithm. x and y

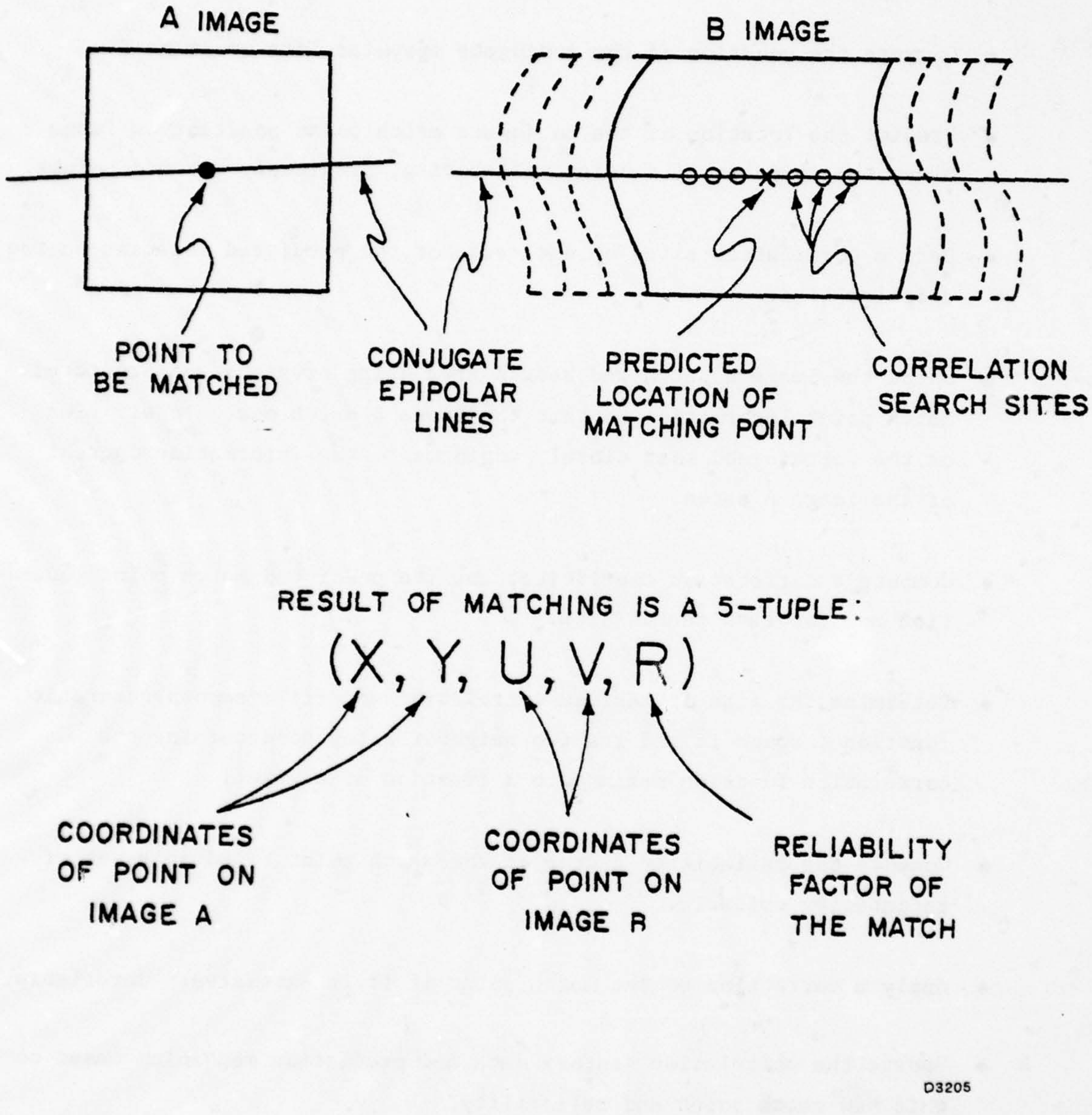


Figure 2-2. Stereo Image Block Matching

are the digital scan coordinates of an evenly spaced grid point on image A. u and v are the digital scan coordinates of the conjugate point on image B: v is actually computed in the epipolar line determination, u is found by the correlation search along the epipolar line. R is the reliability factor of the match, and will be discussed in a later section of this report.

2.1 Correlation Strategy

In determining a match point, it is necessary to place the correlation patch at each site of the image B search area, and accumulate over the patch area the necessary gray-scale sums and cross products needed for the computation of the correlation coefficient. The net result is a value of the correlation coefficient for each site of the search area.

The assumption here is that the image B pixel data that is contained in the search area has already been resampled and shaped according to previous matching history and current predictions. The size of the image A correlation patch and the number of correlation sites on either side of the predicted match point along the search segment are variable, being initially defined as part of the input tuning parameters. The pixel data management scheme using buffer windows is rather straightforward and has been described previously in the First Interim Technical Report (1).

The procedure to avoid in the correlation strategy is the independent accumulation of pixel data for each individual patch placement along the search segment. This scheme results in a great deal of redundant computations, and each image B pixel is accessed repeatedly, once for each patch placement that it is contained in. So, under the current correlation strategy, the basic idea is to access each pixel only once for a given search and to accumulate its gray-scale value when it is available in all the sums and cross products for which it has influence.

The mechanics of the strategy are illustrated in Figure 2-3. Pixels are accessed from the buffers row by row. As each pixel of the image A patch is accessed, its value is accumulated in the patch sum and sum of squares. Likewise, as each pixel of the image B search area is accessed, its value is accumulated in its respective column sum and column sum of squares. Also, cross products are generated for each patch placement along the search segment. When the accumulation of sums and products is complete, the variance of the image A patch is computed; and the array of column sums and the array of

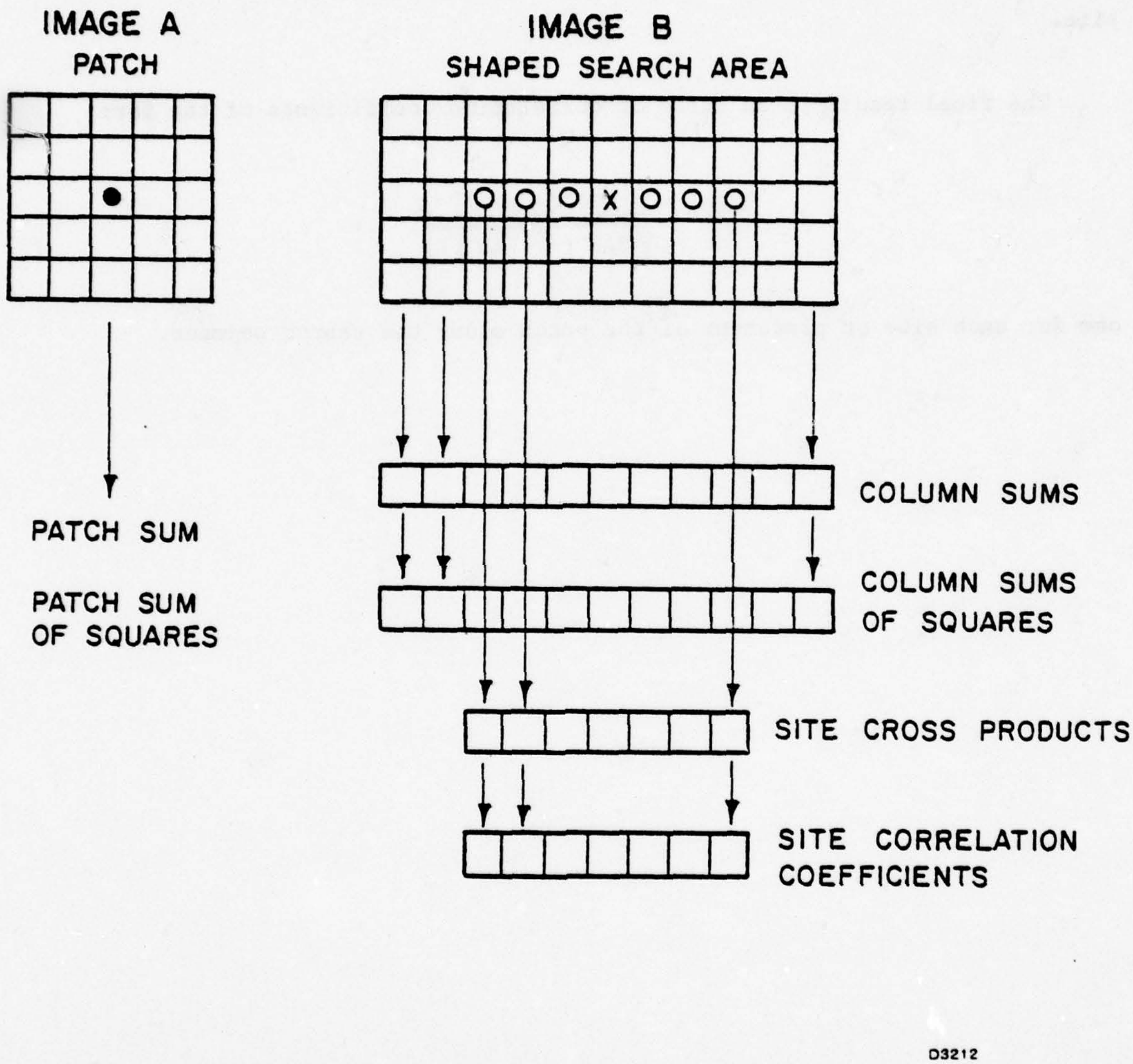


Figure 2-3. Correlation Algorithm

column sums of squares are traversed to generate an image B variance and a covariance for each site of the search segment. The array traversal involves the addition of the next column sum and the subtraction of the last column sum from a running total to simulate the movement of the patch from site to site.

The final result is an array of correlation coefficients of the form,

$$R = \frac{COV\ (A,B)}{\sqrt{VAR\ (A)\ VAR\ (B)}},$$

one for each site or placement of the patch along the search segment.

2.2 Correlation Maximum Determination

The correlation coefficients computed by the above strategy correspond to shaped image B pixel centers. To determine a match point, it is necessary to interpolate the correlation maximum to a fraction of a pixel. This procedure is illustrated in Figure 2-4.

The array of correlation coefficients is searched for the maximum value. Then, by using one value on either side of this maximum, a parabolic curve is fit over the three values. Thus, the point along the search segment at which the derivative of the parabolic correlation function is zero is the point of maximum correlation and is defined to be the match point.

A problem exists when the maximum correlation value at a pixel center occurs at either extremity of the search segment. In this case, a parabolic fit cannot be performed; so the pixel center of maximum correlation is defined to be the match point, and this situation is recorded as part of the reliability factor. The underlying idea here is that this match point is suspect and is a candidate for correction or further processing.

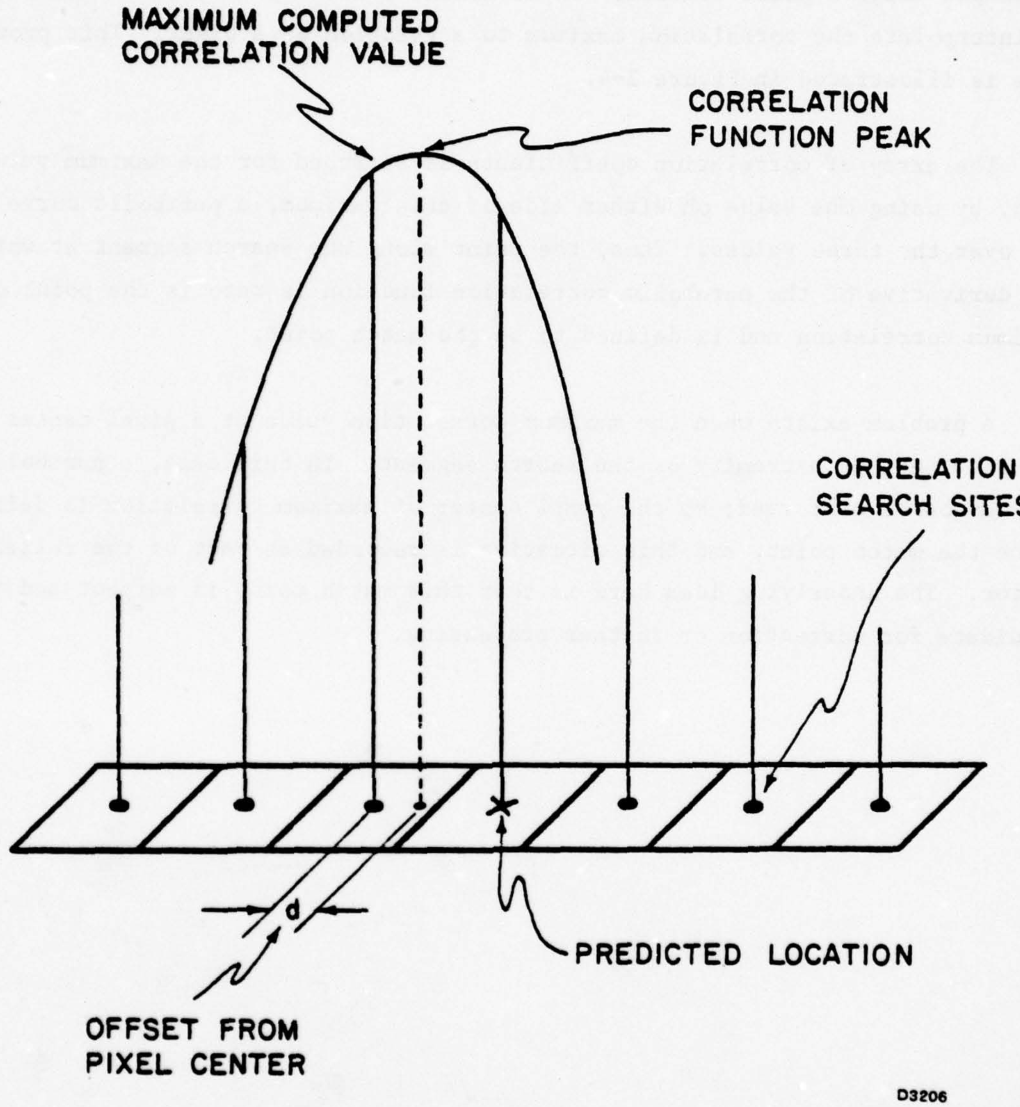


Figure 2-4. Correlation Maximum Determination

3.0 PREDICTION MECHANISM

Epipolar geometry provides the primary constraint on the stereo matching process. This idealized geometry is illustrated in Figure 3-1, showing a number of cases for different orientations of stereo photography. The basic idea is that match points must be on conjugate epipolar lines except when there is excessive film distortion or air refraction. Therefore, this geometry offers a convenient means for making correlation searches one-dimensional along an epipolar line. That is, in forming a match point, the v coordinate is derived strictly from the geometry, thus eliminating the need for a correlation search in a direction normal to the epipolar line.

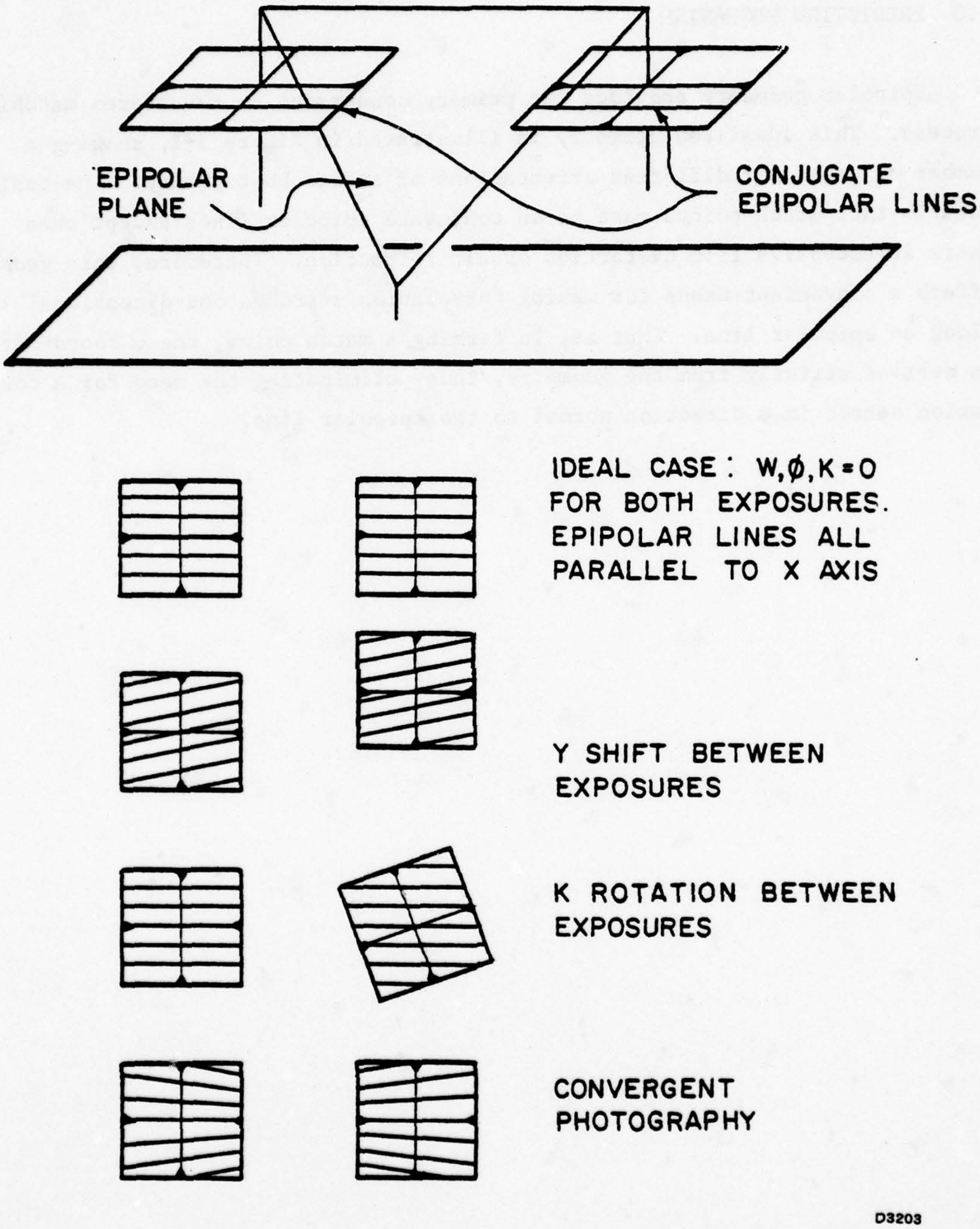


Figure 3-1. Epipolar Geometry

3.1 Epipolar Geometry

The implementation of epipolar geometry in a prediction mechanism requires a knowledge of the relative orientation elements of the stereo pair. These are provided as input parameters. The photogrammetric computations required to produce epipolar predictions on a point by point basis are given in Appendix B. It must be noted that all these computations take place in the photo coordinate systems of the stereo pair, the systems that are functionally related by the relative orientation elements. This is inconsistent with the fact that digital correlation and match point formation occur in the digital scan coordinate systems of the pair. What the algorithm must do, then, to accommodate this inconsistency is to constantly switch back and forth between coordinate systems; making epipolar predictions in photo coordinates, correlating and shaping in scan coordinates, and feeding back match point information in photo coordinates for the next prediction.

This coordinate switching necessitates the use of interior orientation transformations that accurately relate the digital scan coordinate systems to their respective photo coordinate systems. The word accurately here cannot be taken too lightly because it has been shown repeatedly that a great percentage of faulty correlations, and consequently a great percentage of errors in final terrain data, are attributable to inaccurate interior orientation. It is not sufficient to merely digitize the imagery and then process it without a precise mensuration exercise. The point here is that the digitization of an image results in an image transformation that must be modeled. That is, the digital image has its own geometry, incorporating all of the photogrammetric deformations present on the original film as well as distortions created by the digitization process.

A comprehensive procedure for constructing accurate interior orientation transformations is outlined in Appendix A. This procedure, together with relative orientation, is considered a preprocess to the matching algorithm. The transformation coefficients are supplied as input parameters.

3.2 Rate of Change Functions

As explained earlier, epipolar geometry provides a means for predicting the match point v coordinates. But in order to apply this prediction, a prediction for u, the parallax coordinate, is required. This u prediction must primarily take into account the local terrain relief displacement on the images in the area to be correlated. The basic idea is to make this prediction accurate enough so that correlation only has to be performed for one or two pixel sites on either side of the predicted location. The ideal running situation in automatic matching occurs when the tuning parameters are set correctly for the imagery, the image quality and feature content are high, and correlation maxima are found within one pixel of their predicted locations.

A convenient means for characterizing the differential terrain relief displacement between images of a stereo pair is the use of velocity or rate of change functions. This concept is illustrated in Figure 3-2. In the present matching algorithm, the evenly spaced grid to be matched is defined on image A. Therefore, the distance between grid lines of constant x is constant across the image and is termed Δx. The corresponding distances, Δu, on image B are variable and depend on the exposure station positions and orientation and on the slope of the imaged terrain. As can be seen in the figure, the rate of change of image feature placement, $\Delta u/\Delta x$, is one in all cases for flat terrain. Then, $\Delta u/\Delta x$ is less than one or greater than one depending on whether the terrain slopes toward or away from exposure station A.

Analytically, the relationship between the velocity function $\Delta u/\Delta x$ and the actual terrain slope $\Delta h/\Delta X$ is:

$$\frac{\Delta u}{\Delta x} = 1 - \frac{B \dfrac{\Delta h}{\Delta X}}{H + X, \dfrac{\Delta h}{\Delta X}}$$

or inversely,

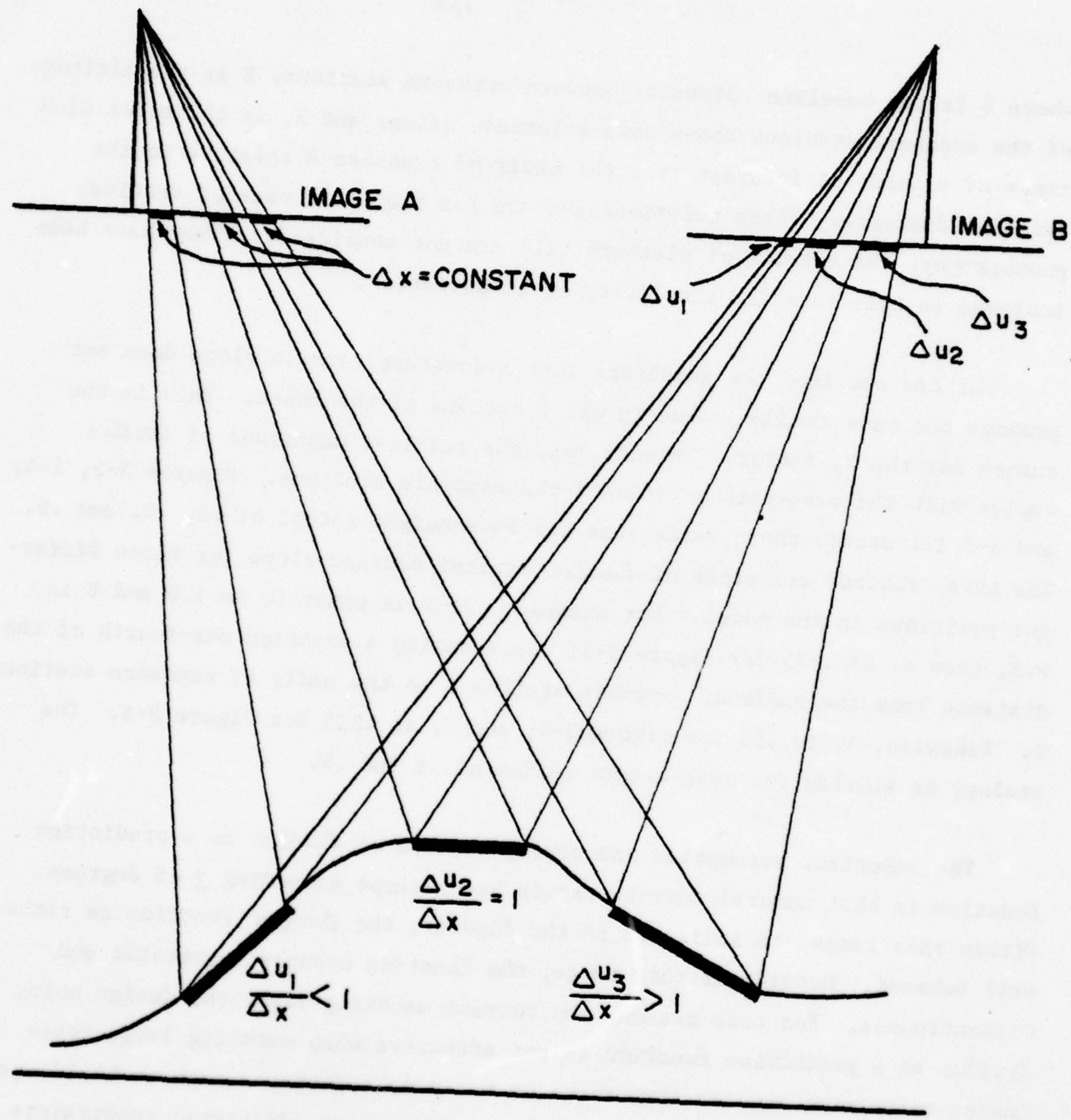


Figure 3-2. Rate of Change Function: $\frac{\Delta u}{\Delta x}$

$$\frac{\Delta h}{\Delta X} = \frac{H\,(1 - \frac{\Delta u}{\Delta x})}{B - X_1\,(1 - \frac{\Delta u}{\Delta x})}$$

where B is the baseline distance between exposure stations, H is the altitude of the exposure stations above some reference datum, and X_1 is the model distance of a point of interest from the nadir of exposure A relative to the baseline distance. These relationships are for the ideal case of vertical photography; the effects of platform tilt are not considered. They have been included in this form for simplicity of illustration.

One can see from the equations that a constant terrain slope does not produce the same $\Delta u/\Delta x$ value in all locations of the model. This is the reason for the X_1 factor. In addition, the relative magnitude of $\Delta u/\Delta x$ varies with the base-height ratio of the exposure stations. Figures 3-3, 3-4, and 3-5 illustrate these variations for base-height ratios of .3, .6, and .9. The three figures are plots of $\Delta u/\Delta x$ against terrain slope for three different positions in the model. For example: If H is taken to be 1.0 and B is 0.3, then X_1 is .075 for Figure 3-3; representing a position one-fourth of the distance from the nadir of exposure station A to the nadir of exposure station B. Likewise, X_1 is .15 for Figure 3-4, and X_1 is .225 for Figure 3-5. The analogy is similar for base-height ratios of .6 and .9.

The important assumption underlying the use of $\Delta u/\Delta x$ as a prediction function is that natural terrain rarely has a slope exceeding $\pm$ 45 degrees. Within this range, as indicated in the figures, the $\Delta u/\Delta x$ function is rather well behaved. Outside of this range, the function becomes asymptotic and discontinuous. For this reason, the current matching algorithm design using $\Delta u/\Delta x$ as a prediction function is not effective when matching large scale images containing angular structures or man-made structures such as buildings, towers, storage tanks, etc. These situations require additional constraints on the $\Delta u/\Delta x$ function or an entirely new prediction scheme. One alternative is to shift the evenly spaced matching grid from the image A coordinate system to the model coordinate system. In this way, image feature velocities can be

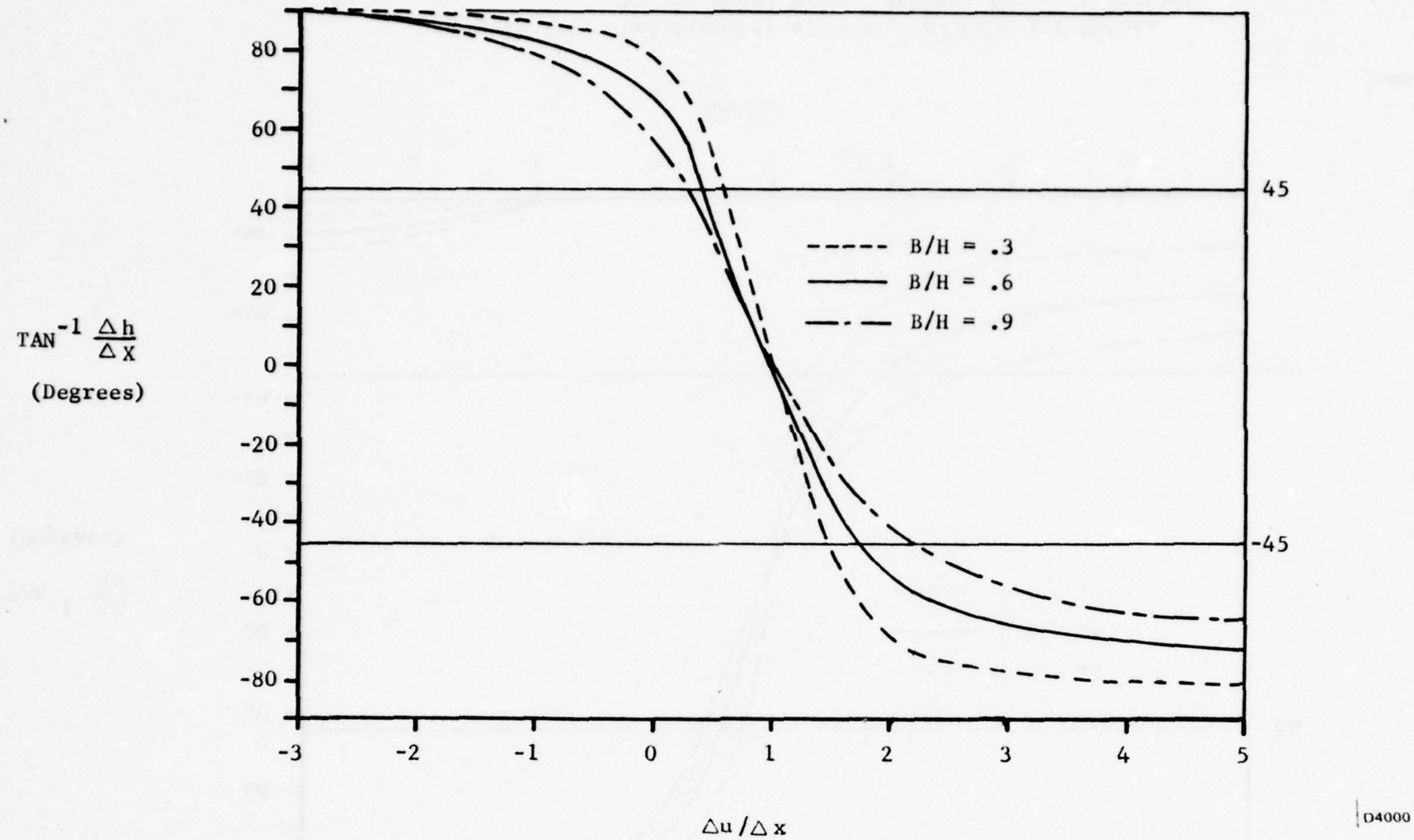


Figure 3-3 Relationship of $\Delta u/\Delta x$ to $\Delta h/\Delta X$ for Points in The Model Between Exposure Station A and The Center of The Model

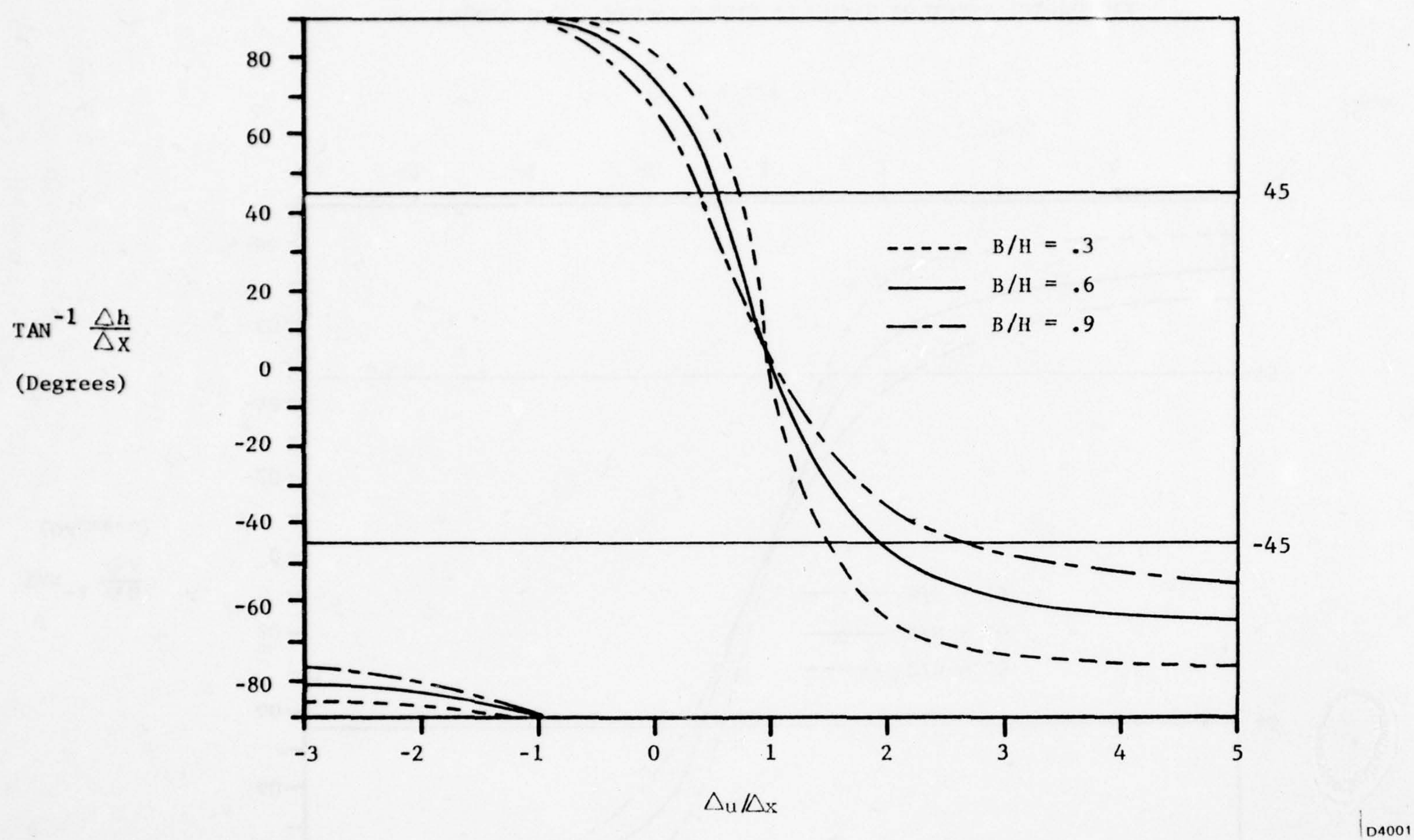


Figure 3-4 Relationship of $\Delta u/\Delta x$ to $\Delta h/\Delta X$ for Points in The Model Midway Between Exposure Station A and Exposure Station B

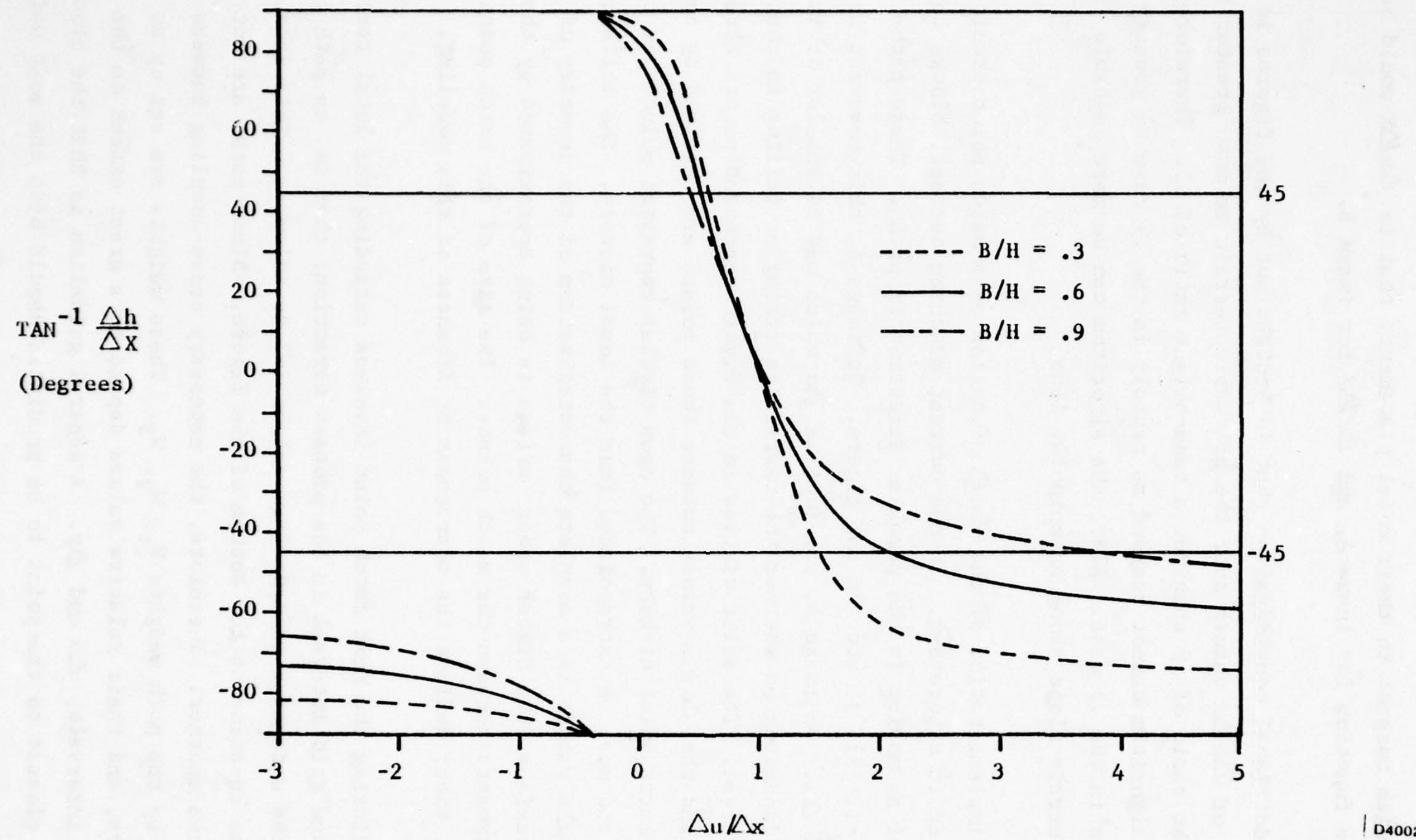


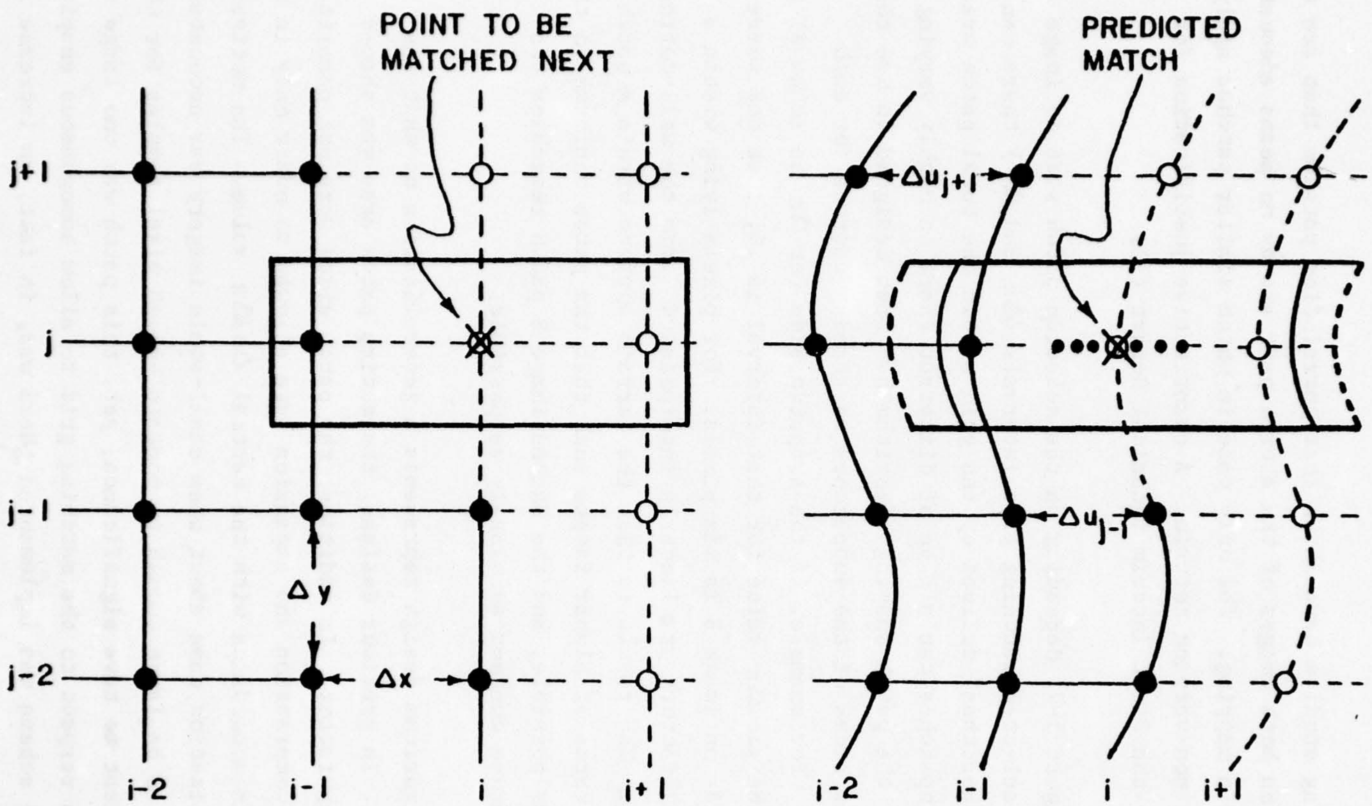
Figure 3-5 Relationship of $\Delta u/\Delta x$ to $\Delta h/\Delta X$ for Points in The Model Between The Center of The Model and Exposure Station B

measured with respect to their model placement; that is $\Delta x/\Delta X$ would be the prediction function for image A, and $\Delta u/\Delta X$ for image B.

An additional consideration that is brought out by the figures is that the range of $\Delta u/\Delta x$ values over the allowable terrain is much greater for a base-height ratio of .9 than for a base-height ratio of .3. Therefore, the matching algorithm cannot respond as rapidly in the .9 case to greatly varying terrain as in the .3 case. Also, the algorithm can be more unstable in tracking the terrain slope along an epipolar line.

The implementation of the $\Delta u/\Delta x$ function as a match point predictor is illustrated in Figure 3-6. In the current matching concept, blocks can be thought of as moving in the parallax direction in paths. These paths are labeled j-2, j-1, j, etc. in the figure. On image A, this movement is in jumps of Δx. On image B, the jump is Δu which may be smaller or larger than Δx depending on whether the terrain is rising or falling in that particular interval. The solid circles in the figure represent points already matched, and the $\Delta u/\Delta x$ values between these points are retained by the algorithm as match point history. The open circles represent points yet to be matched, and must be extrapolated from the local history. The collection of these $\Delta u/\Delta x$ values is a complete characterization of the geometry of the terrain surface; the higher order surface is being approximated by the small linear segments between the match points. The size of the match point grid interval, then, defines the coarseness or fineness of this modeling.

Predicting the next match point involves extending the local terrain surface one grid interval in the process direction; that is on path j in the figure, the extension is made from column i-1 to column i. This is carried out by the equation at the bottom of the figure. Block paths are not independent of one another. Therefore, the necessary cross-coupling between paths is achieved by the path weights W_1, W_2, W_3. These weights are set up as tuning parameters, and their relative values depend to a great extent on the size of the grid intervals, Δx and Δy. A general guideline is that the block path interval closest to the point to be predicted should have the most weight.



$$u_{i,j} = u_{i-1,j} + \left[W_1 \left(\frac{\Delta u}{\Delta x} \right)_j + W_2 \left(\frac{\Delta u}{\Delta x} \right)_{j-1} + W_3 \left(\frac{\Delta u}{\Delta x} \right)_{j+1} \right] \text{WHERE } W_1 + W_2 + W_3 = 1$$

D3207

Figure 3-6. Match Point Prediction by Use of $\frac{\Delta u}{\Delta x}$ Function.

3.3 Patch Shaping and Resampling

Previous matching studies have shown that correlation patches that are the same size and shape on both images of the stereo pair are by no means adequate for performing terrain mapping. The only case in which similar patches apply is the case of flat, non-varying terrain. A quantitative justification for this can be found in the First Interim Technical Report (1).

Referring to Figure 3-7, depending on the selected patch size on image A with respect to the selected matching grid intervals (Δx and Δy) there can be any number of "subpatches" defined by the grid within the total patch area. On image B, these subpatches can all be of different shape in highly varying terrain. Therefore, the patch shaping algorithm has been designed to use the $\Delta u/\Delta x$ values on the sides of the subpatches as shaping factors for each individual subpatch. For example, if the subpatch side (or Δx on image A) is ten pixels, and if the $\Delta u/\Delta x$ value for that interval is .6, then the corresponding subpatch side on image B is six pixels. For pixels lying within a subpatch, the shaping factors are linearly interpolated from the well-defined subpatch side values. The result is that the terrain surface within a patch is being modeled in terms of planar facets such that the patch conforms to the terrain as closely as possible, and the shaped image B patch resembles the image A patch in feature content as closely as possible.

This shaping algorithm design represents a generalization of what was reported previously. In previous designs, the entire patch area was shaped with the same shaping factor. In addition, the patch sides were not permitted to slant; and patch compression and expansion were allowed to occur only in the parallax direction in accordance with the central $\Delta u/\Delta x$ value. The motivation for the generalization came about when small-scale imagery was encountered. Here, the patch had to be large enough to contain enough pixel samples for the correlation coefficient to have significance; yet, this patch was too large in ground distance with respect to the matching grid to allow homogeneous shaping. When the generalized scheme was implemented there was, in fact, an increase in the values and reliability of the correlation coefficients on this small-scale

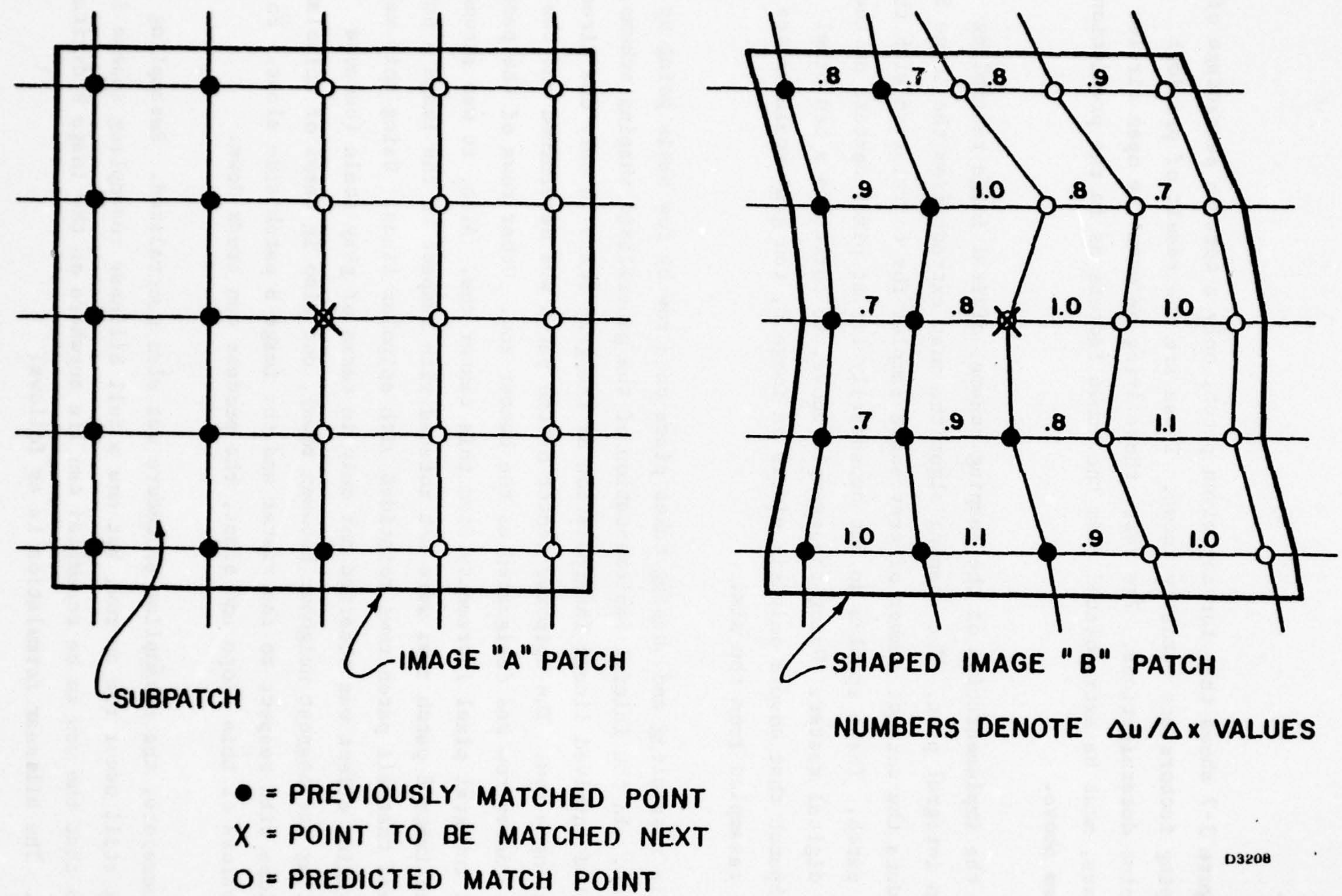


Figure 3-7. Correlation Patch Shaping

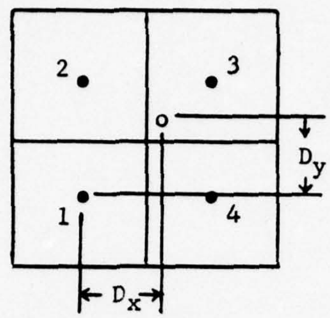
imagery.

Figure 3-7 shows that for any given patch, only a certain percentage of the shaping factors are actually known. These are the result of previous match point determinations. The rest, those lying between the open circles in the figure, must be extrapolated from the known factors as in the prediction mechanism above.

In the implementation of the shaping scheme, digital image resampling plays an integral part. The shaping algorithm must extract from the image B buffer data the correct number of gray scale samples for correlation with the image A patch. These samples do not necessarily lie at pixel centers on the image B digital raster. Using the example above, to correlate a ten pixel wide subpatch that covers only six pixels on image B, ten gray scale values must be resampled from the six.

This resampling and shaping takes place on a row by row basis going up the patch. In the initial implementation of the generalized shaping scheme, the method involved linear interpolation of the gray scale in only one direction, along a row. The epipolar center of the patch was determined and the nearest raster row was designated as the center row. Other rows of the patch fell at integral pixel increments from this center row. Also, it was assumed that the image B patch rows were not rotated with respect to the image A patch rows, and that all patch rows coincided with epipolar lines. Using this method, an averaging effect was observed not only in terms of gray scale (because resampling was nearest neighbor between rows), but also in terms of epipolar line slope with respect to the raster and the image B patch side slant. For large values of this slope and slant, the process can break down.

Therefore, the resampling procedure was also generalized. Resampling and shaping still occur row by row, but now a full bilinear resampling scheme is used so that the row to be resampled can lie anywhere on the image B digital raster. The bilinear formulation is as follows:



Consider the four pixels shown above whose centers are labeled 1, 2, 3, 4. The resampling task is to assign a gray scale value to the point whose distance components from pixel 1 are D_x and D_y, where D_x, $D_y < 1$. If G_i $(i = 1, 4)$ are the gray scale values assigned to the pixel centers, then the desired value G is give by:

$$G = G_1 (1 - D_x) (1 - D_y) + G_2 (1 - D_x) D_y + G_3 D_x D_y + G_y D_x (1 - D_y).$$

In moving this four-pixel resampling window to correspond to a patch row, the transition from subpatch to subpatch is made using finite difference techniques.

After the generalized shaping and resampling concepts were implemented, the reliability of the matching over approximately 1,400 match points increased from 80% to 83%. This will be discussed in the next section.

4.0 ALGORITHM CONTROL MECHANISMS

The use of the correlation coefficient as the similarity metric in automatic matching systems gives rise to several pitfalls that must be dealt with. Some of these pitfalls are false correlation maxima, low correlation values with false peaks in featureless areas due to the low signal to noise ratio, and correlation dropout due to gross noise or dissimilar imagery. Therefore, it is necessary for the algorithm to monitor itself and apply match point corrections where appropriate, to overcome the possible instability of the correlation coefficient.

4.1 Reliability Factor

In an effort to allow for match point corrections, a reliability factor has been designed into the algorithm so that both the algorithm itself and the analyst running the algorithm can have an indication of how the matching is progressing. Referring to Figure 4-1, a number of reliability criteria have been set up such that the N digit reliability factor contains one digit for each criterion. In the current algorithm design, there are five criteria; but this number can be increased as more or different processs characterization functions come into use. There is a hierarchy among the criteria such that the most important ones occupy the most significant digits of the reliability factor. A reliability factor is generated for each point matched.

The first criterion is the value of the correlation coefficient. If this value falls below a certain threshold value that was initially input to the process as a tuning parameter, then the digit for this criterion in the reliability factor becomes a one. Otherwise, it is zero. Likewise, the standard deviation of the gray scale samples over the patch are evaluated. In a previous implementation the image A standard deviation was measured for this criterion. Now, the algorithm also generates an image B patch standard deviation and also the difference between these standard deviations. An excessive difference here suggests image dissimilarity which can be caused by feature-sized noise, such as film scratches and dust specks, or by occluded terrain in the case of large base-height ratios.

The third criterion is the flag that was mentioned in Section 2.2 indicating that the correlation maximum was encountered at the extremity of the search segment. The next criterion indicates whether the $\frac{\Delta u}{\Delta x}$ value for the current match point is outside the range for allowable terrain. Again, the allowable threshold is set as a tuning parameter. The last criterion deals with a judgement as to the quality of the correlation peak. Sometimes this is a rather vague indicator because for large, reliable patches the slope of the correlation peak is generally small; yet, for small patches where false peaks are prevalent, the slope can be rather large.

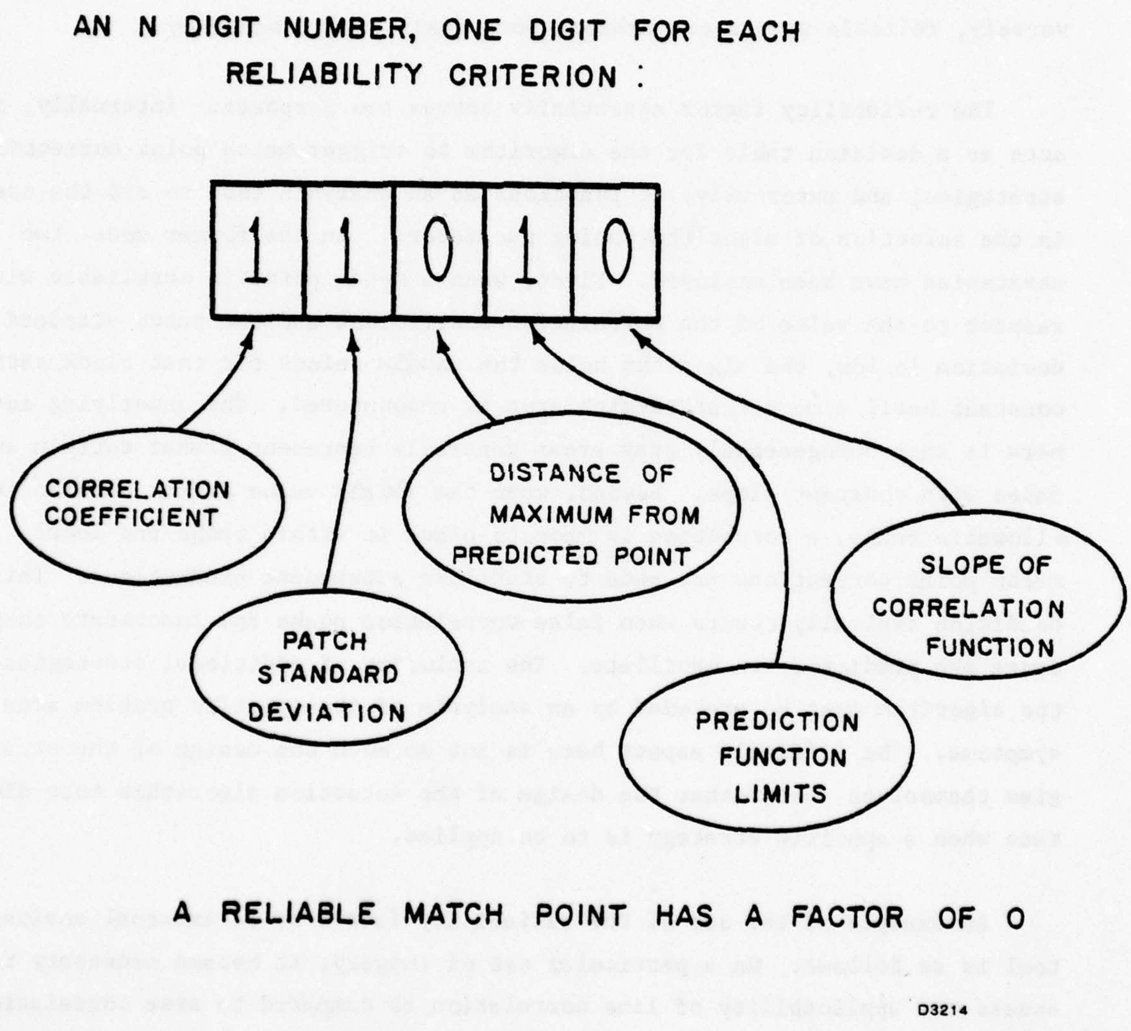


Figure 4-1. Reliability Factor

A match point, then, that is reliable with respect to all of these criteria has a reliability factor of zero. However, the reliability factor is more a general indicator of the stability of the matching process than of the absolute match point accuracy. In most cases, instability and inaccuracy are well correlated. There are however, cases where match points that have been deemed unreliable during matching are very accurate positionally, and conversely, reliable points can exhibit some positional inaccuracy.

The reliability factor essentially serves two purposes. Internally, it acts as a decision table for the algorithm to trigger match point correction strategies; and externally, it functions as an analysis tool to aid the user in the selection of algorithm tuning parameters. In the former mode, two strategies have been employed. First, when a match point is unreliable with respect to the value of the correlation coefficient and the patch standard deviation is low, the algorithm holds the $\Delta u/\Delta x$ values for that block path constant until a more feature-rich area is encountered. The underlying idea here is that homogeneously gray areas generally represent planar terrain surfaces with constant slope. Second, when the $\Delta u/\Delta x$ value drifts outside its allowable range, a correction is made to place it within range and local match point corrections are made to stabilize subsequent predictions. This condition typically occurs when false correlation peaks and inaccurate shaping cause the predictor to oscillate. The inclusion of additional strategies in the algorithm must be preceded by an analysis of the specific problem area symptoms. The difficult aspect here is not so much the design of the strategies themselves, but rather the design of the detection algorithms that dictate when a specific strategy is to be applied.

An example of the use of the reliability factor as an external analysis tool is as follows: On a particular set of imagery, it became necessary to assess the applicability of line correlation as compared to area correlation. An image area containing 1,476 match points was chosen, and the matching grid intervals were 2 pixels in the y direction and 5 pixels in the x direction. The search segment length included 3 pixels on either side of a predicted point. For the line correlation case, the patch size was set to be 1 by 35

pixels. This corresponds to a single row of pixels on an epipolar line. For the area correlation case, the patch size was 3 by 35 pixels, giving three times as many samples. The reliability summaries printed at the end of each matching run appeared as follows:

Line Correlation		Area Correlation	
Criterion	Percent	Criterion	Percent
1	3.8	1	3.0
2	1.2	2	.3
3	21.4	3	11.9
4	13.5	4	11.9
5	8.0	5	9.6
Total Reliable	69.8		75.7

The percentages for the five reliability criteria represent the unreliable match points. The bottom line percentages represent total reliability. The conclusion here is that the averaging effect of the aerial patch lends more stability to the process. Line correlation is rather jittery. This is borne out by the fact that the percentage of matches where the correlation maximum was found at the extremity of the search segment (criterion 3) halved itself when going from line correlation to area correlation. The increase in reliability of the values of the correlation coefficients and standard deviations (criteria 1 and 2) occurs primarily because of the increased number of samples in the area correlation. The increase in unreliability with respect to the slope of the correlation function can be attributed to the fact that larger patches typically have flatter correlation peaks.

4.2 Wandering Block Tolerance

A control and correction mechanism that plays an important part in keeping the matching algorithm running through difficult image areas is the wandering block strategy. In hard-to-correlate image areas and in areas where false correlation peaks predominate, it is typical for blocks on some paths to lag behind or to jump ahead with respect to neighboring, more reliable block paths. The result is that the $\Delta u/\Delta x$ function produces artificial dips and bulges in the terrain surface at these match points.

The wandering block tolerance is a tuning parameter that has been set up to detect and correct these situations. When each column of blocks has been matched (blocks whose centers all have the same X coordinate on image A), the algorithm stops matching to analyze the entire column for wandering blocks. The u coordinate position of each block on image B is compared to the average u position of its two nearest neighboring blocks. These concepts are illustrated in Figure 4-2. If the block distance from the average position exceeds the preset tolerance, then the match point for that wandering block is moved toward the neighbor average by a weighted amount. This weight is also entered as a tuning parameter. For example, if the weight is 1.0, the match point is corrected to the neighbor average. If it is .5, the point is corrected halfway to the average, etc.

The strategy here is set up such that blocks which have been flagged as unreliable are corrected first. Then the remaining blocks of the column are analyzed. The value of the tolerance distance must typically be consistent with the allowable terrain slope limits set up for the run. When corrections are applied to wandering blocks, the matching history in terms of the stored $\Delta u/\Delta x$ values is also altered to account for the corrections.

It can be observed that the setting of the wandering block tolerance and the correction weight act as a control on the sensitivity of the matching algorithm. A large value of the tolerance makes the block paths more independent of one another and more responsive to small terrain fluctuations. On

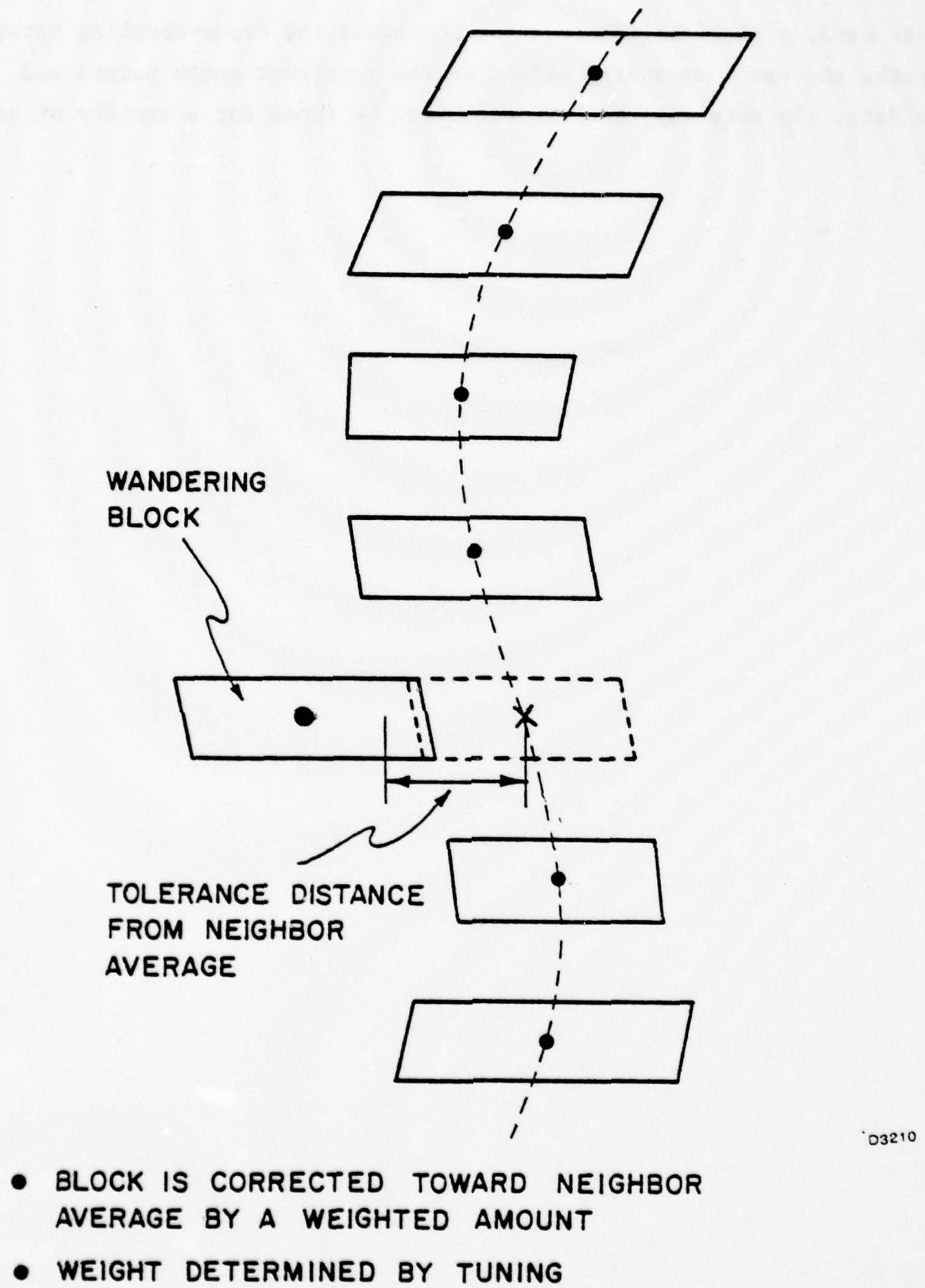


Figure 4-2. Wandering Block Tolerance

the other hand, a small tolerance value increases the cross-coupling between block paths and has a smoothing effect on the resultant match points and terrain data. In this way, the algorithm can be tuned for a variety of situations.

4.3 Patch Center Shift

There is no hard and fast rule in stereo matching that dictates that a match point must be determined for the center of a correlation area. Referring to Figure 4-3 and the figures in Section 3.0 regarding prediction and shaping, there is always a certain percentage of the patch area whose shape is well-known and a percentage whose shape is estimated. The area of known shape is the area in which some match points have already been determined. These known and unknown patch areas can be increased or decreased with respect to one another by shifting the nominal correlation center. For example, shifting the center ahead or to the right of the patch, as pictured in Figure 4-3 increases the areas of known shape. Therefore, the correlation is biased to a greater extent by gray scale samples that are presumably known to correlate well.

Theoretically, this increase in known area should allow the matching algorithm to proceed through highly varying terrain and up steep slopes in a more cautious manner since the patch is well-grounded in familiar area. Conversely, if the unknown area is increased, the algorithm is more liberal in its perception of the terrain and the chances of mismatching increase.

The algorithm currently has the capability for shifting the nominal center in whole grid intervals anywhere within the patch area. However, a complete analysis has not been made to date to assess the effectiveness of the shift for various image events. For all the imagery that has been processed, the correlation center has coincided with the geometric center of the patch.

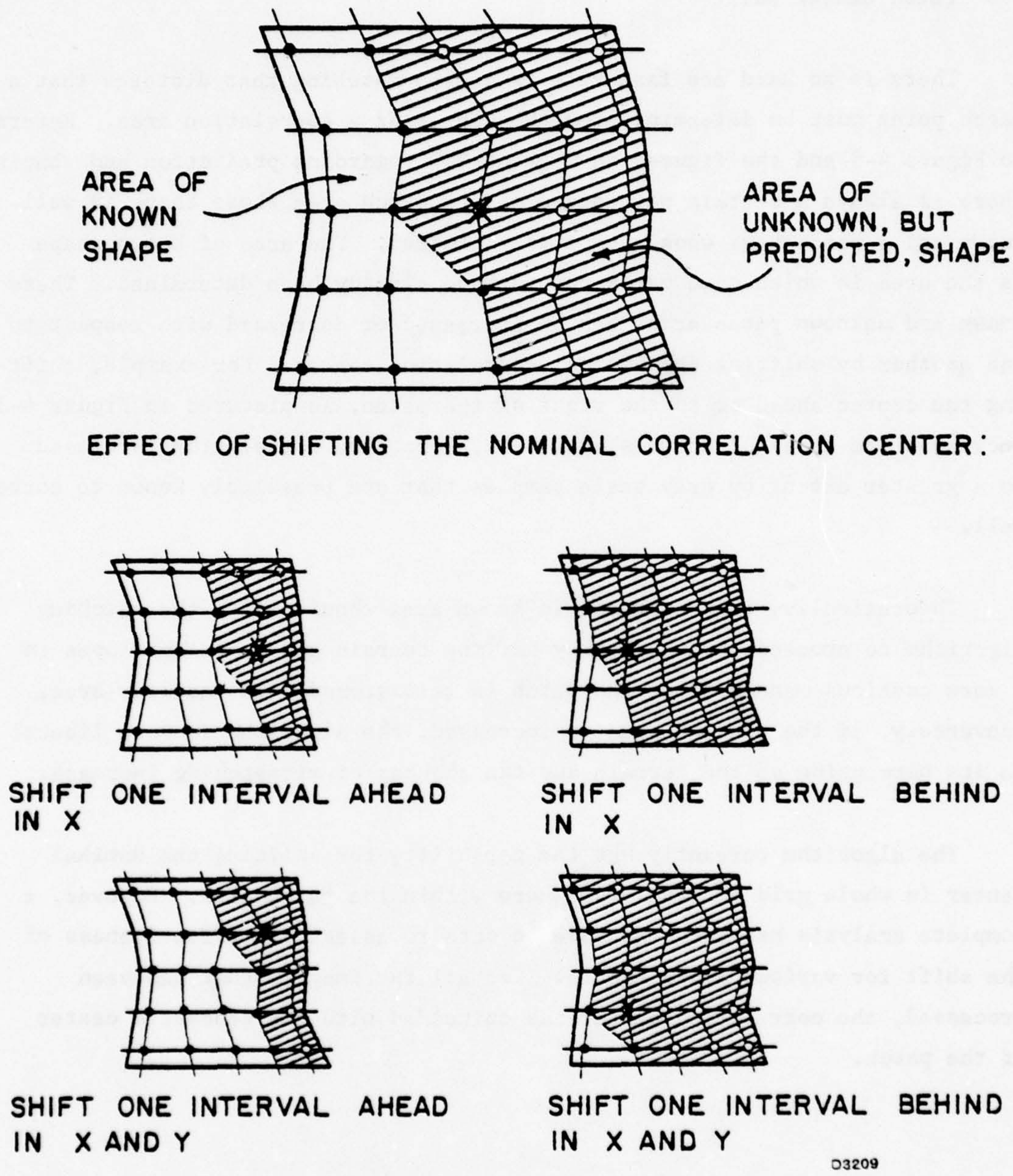


Figure 4-3. Nominal Correlation Center Shift

5.0 ADDITIONAL CAPABILITIES AND GENERALIZATIONS

When the matching algorithm was originally designed, a particular set of test data was used to verify the design concepts. But the complete range of stereo conditions is not exhibited in any one stereo pair. As a result, the algorithm remained untested for certain cases. Also, there are cases that fall completely out of the range of the algorithm's capability. This section of the report describes both the steps that were taken and the steps that need to be taken to provide extended algorithm capability over the range of possible cases.

5.1 Non-parallel Epipolar Lines

In the original design of the stereo matching algorithm, epipolar lines were assumed to be essentially coincident with the rows of pixels on the rasters of both image A and image B. This is usually not the case with real imagery. However, the imagery can always be digitally rectified by a preprocess such that this condition always holds. But the rectification requires image resampling which is rather costly in time, and which can produce a slight amount of image degradation which is magnified by the fact that the image must be resampled again in the algorithm shaping process. So for the current algorithm implementation, the decision was made to resample only once (during shaping) and to generalize the algorithm to take into account both non-parallel epipolar lines and epipolar lines that do not coincide with digital raster lines.

Figure 5-1 illustrates a general situation. The epipolar lines are sloped with respect to the image A digital raster because of the image B platform translation. The digital raster on image B is not parallel to the image A raster or to the epipolar lines because of the platform kappa rotation. As expected by the current algorithm design, the evenly spaced match point grid is parallel to the raster of image A. The important point to consider is that the correlation subpatches that constitute a patch on image B are now no longer trapezoidal as in the previous discussions, but rather are generalized quadrilaterals. The slant of the subpatch sides is determined by the local $\Delta u/\Delta x$ values that are generated as match points slide along the slanted epipolar lines.

The generalized algorithm design handles this situation in the following way. Match point predictions are made along epipolar lines, although the correlation patch is oriented parallel to the digital raster on image A. In this way, the image A patch need not be resampled. This is a justifiable approach as long as all corresponding points within the image A and image B

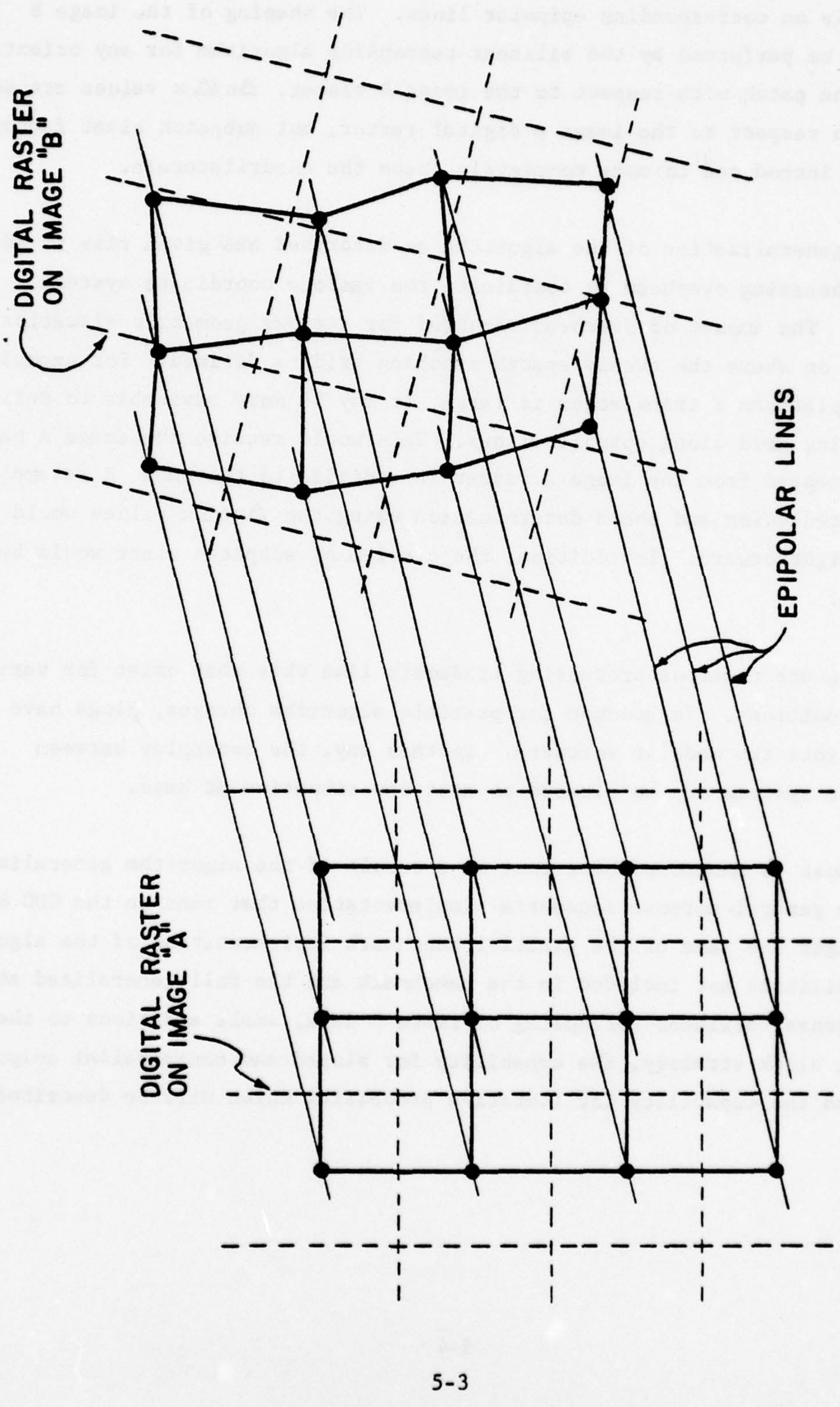


Figure 5-1. Match Point Configuration for Exposures that Are Relatively Translated in Y and Rotated in Z

patches lie on corresponding epipolar lines. The shaping of the image B patch can be performed by the bilinear resampling algorithm for any orientation of the patch with respect to the image B raster. $\Delta u/\Delta x$ values are computed with respect to the image A digital raster, but subpatch slant factors have been introduced to more completely shape the quadrilaterals.

The generalization of the algorithm as described has given rise to additional processing overhead to coordinate the various coordinate systems involved. The amount of overhead required for various geometric situations depends on where the evenly spaced matching grid is defined. For example; when the platform Y translation is large, it may be more advisable to define the matching grid along epipolar lines. This would require the image A patch to be resampled from the image A raster in addition to the image B resampling, but the prediction and shape determination using the $\Delta u/\Delta x$ values would be more straightforward. In addition, the problem of subpatch slant would be minimized.

There are numerous processing tradeoffs like this that exist for various stereo conditions. To account for possible algorithm changes, plugs have been designed into the modular software. In this way, the interplay between coordinate systems may be altered to meet the situation at hand.

It must be mentioned here that as a result of the algorithm generalization, the general-purpose sequential implementation that runs on the CDC 6400 is no longer the same as the parallel benchmark implementation of the algorithm. The capabilities not included in the benchmark are the full generalized shaping by subpatches, bilinear resampling of image B data, small additions to the wandering block strategy, the capability for sloped and non-parallel epipolar lines, and the capability for iterative processing which will be described next.

5.2 Iterative Processing

A new option has been designed into the matching algorithm to allow images to be matched by iterative refinement. That is, subsequent passes of the algorithm over the same image area make use of the match point information generated on a previous pass. The observed net result is that, on subsequent passes, reliable match points tend to stay the same but unreliable areas are improved.

When this iteration option is called for, the prediction mechanism cf the algorithm is turned off. The match points from the previous iteration are the actual predictions. What this scheme does, then, is eliminate the patch areas of unknown shape that were described above in Section 3.3. Thus, match points are generated using neighboring match points that are presumably more locally valid in all directions from the desired point.

Using the reliability summary as a judgement factor, it has been observed that the total match point reliability can be increased by as much as ten percent under the iteration option. The largest increase in reliability occurs at the first iteration as is to be expected. After the first, subsequent iterations yield a few more percentage points. However, after the fourth iteration, the reliability stays the same or in some observed cases actually decreases slightly. In these cases, the algorithm tuning was the same for all iterations. A thorough analysis of this phenomenon has not been made, but it is reasonable to assume that after a certain point the process is oscillating on unreliability- merely changing the positions of unreliable match points- and unable to improve without additional information or different tuning.

In attempting to improve the reliability the problem of tuning the algorithms for subsequent iterations to achieve maximum refinement is encountered. Again, this has not been thoroughly investigated. For example, should the correlation patch be smaller or larger on subsequent iterations? A smaller patch should theoretically offer more accuracy in the refinement,

but a larger patch would take into account more surrounding match point data, yet would have a smoothing effect. There are good arguments for either alternative that apply over the whole range of tuning parameters.

5.3 Algorithm Tuning

All previous discussions have hinted that the way the algorithm perceives imagery and terrain cay be changed by changing a set of control parameters that are referred to as tuning parameters. The motivation behind this design is that the algorithm must behave differently for different image and terrain events that can occur. The nature of these different events is determined by the type of sensor used, the stereo taking conditions, and the highly variable characteristics of the terrain that is imaged.

An experiment was conducted on a particular set of imagery to understand the effects of tuning changes. Fifty (50) matching runs were made on the same area of imagery that contained approximately 1,000 match points. The tuning parameters were varied one by one from one run to the next in an effort to constantly increase the matching reliability. The iteration option was not employed for this experiment. The result was that the matching reliability for all the tuning cases ranged from a minimum of 49% to a maximum of 82%.

Each tuning case changes the behavior of the algorithm to certain types of terrain. It was observed that small changes to the values of the tuning parameters sometimes made very large changes in algorithm behavior. The best matching seems to occur when the process is tuned such that it is on the brink of instability. In this way, the algorithm is most responsive to terrain variation, yet stable enough to track the image accurately from one end to the other.

A major factor in determining how the algorithm responds is how the algorithm starts out cold on an image. Experimental results show that the effects of inaccurate matching at the start can be observed as much as 100 scan lines down the image. There are essentially two alternatives to getting the algorithm started. The first, and most accurate, is to supply as input the starting positions and initial terrain slopes of each block path.

The second is to supply one or more manually measured points from which the algorithm can extrapolate the starting match point positions. The latter is a desirable approach from a user standpoint because the measurement of many closely spaced points and the estimation of terrain slope is a rather tedious and time-consuming operation. However, this automatic approach can be error prone depending on the number of points actually measured and the local terrain variation present in the vicinity of the starting block positions.

The automatic acquisition technique performs a matching iteration over extended search lengths. From the few measured points that are supplied, the block centers for the first column of match points are linearly extrapolated. A correlation is performed at each of these block centers using rather long search lengths to account for maximum parallax differences. There is no patch shaping involved here because the local $\Delta u/\Delta x$ values are unknown. It is as if flat terrain were being processed. Then predictions are made for the next column of match points, and they are correlated similarly. At this point, the $\Delta u/\Delta x$ values are computed between these first two columns and the process backs up to start over this time shaping as it goes. second pass acts as a refinement so that at its completion, all the starting positions and shapes are available and control is transferred to the full matching algorithm which starts once again at the first column. In the acquisition, as each match point is correlated repeatedly, the search segment size is decreased progressively.

For summary purposes, the following list of matching algorithm input values is given. The distinction is made between parameters that are constant for a particular stereo pair and ones that are part of the tuning parameter set.

<u>Input Parameters</u>

- Relative orientation elements - these specify the baseline distance and relative tilt angles that relate exposure station B to exposure station A.

- Interior orientation transformations and camera focal length - these specify the conversion between digital scan coordinates and photo coordinates.

- Image buffer parameters - a characterization of available buffer sizes and starting line addresses.

Tuning Parameters

- Match point grid limits and interval sizes.

- Correlation patch size in two dimensions.

- Number of correlation sites along search segment.

- Amount of nominal patch center shift.

- Block path prediction weights.

- Wandering block tolerance and distance correction factor.

- Reliability thresholds for the correlation coefficient, standard deviation, $\Delta u/\Delta x$ range, and slope of the correlation function.

- Prediction option selector.

- Manually measured start-up points.

5.4 The Matching of Non-central Perspective and Dissimilar Imagery

All of the algorithm development and implementation described thus far has taken place for central perspective photography. The epipolar prediction geometry is well-defined and straightforward with respect to this kind of imagery. If the matching algorithm is to be applied to non-central perspective imagery, then the module of the algorithm that requires redesign is the prediction mechanism. All other modules can essentially remain the same. In order to achieve accurate predictions, the dynamics of the sensor of interest must be modeled in such a way that conjugate points on the images lie on pseudo epipolar lines. The motive here is to keep the correlation search one-dimensional. The modeling problem involves finding the direction on the imagery in which terrain relief displacement occurs. Unlike epipolar lines, this direction may be different for the two images of the stereo pair and also for different regions in the same image. But if the terrain relief is analytically predictable in small localized areas, there is a good chance for matching success. To date, processing of non-central perspective imagery has not taken place with the current matching algorithm.

Between two images that have the same ground coverage, image dissimilarity can occur in two domains: the geometric domain, and the radiometric or intensity domain. In the geometric domain, image dissimilarity between images to be matched is attributable either to an overall scale difference or to differing sensor dynamics. Scale dissimilarity can be handled by the current algorithm. As long as the relative orientation elements and focal lengths reflect the scale change, correct scale matching is a straightforward by-product of the resampling and shaping schemes. The larger scale image may have to be filtered in some cases to reduce the high frequency noise and to make its intensity more similar to that of the smaller scale image, but this is the only additional processing that would be required. When image dissimilarity occurs in an overlapping pair because different sensors are used, the remarks of the previous paragraph apply. The matching algorithm must model the sensors such that accurate geometric predictions can be made.

In the intensity domain, two alternatives exist for solving the problem of image dissimilarity. The first alternative follows the assumption that all the matchable features are contained in both images, but under differing gray-scale renditions. The approach to take here is a form of intensity shaping which is analogous to the geometric patch shaping contained in the present algorithm. The gray-scale differences are modeled according to a gray-scale mapping function such that when resampling occurs for image B, for instance, the pixel gray values are transformed by the mapping function into the equivalent image A values. Thus, the correlation algorithm can operate on seemingly similar imagery. But when this intensity modeling is not possible or when features are not common to both images, as in the case of radar vs photography, then a second alternative must be employed. Both images must be transformed either by preprocessing or through resampling to a third, common intensity spectrum which enhances the similarities and suppresses the dissimilarities. One such image transformation is the feature edge or gradient operation. Again, the general principle is that automatic image matching can only occur if the two dissimilar images can be made to appear similar to the matching metric through some sort of modeling or intensity transformation.

6.0 PROCESSING EXAMPLE

To illustrate the behavior of the stereo matching algorithm described in this report, a stereo pair taken over the Phoenix - South Mountain area in Arizona was processed. The scale of the photography is 1:48000, the focal length of the mapping camera being nominally six inches. Two by two inch sections from the original photos were digitized at ETL using a 35 micrometer scanning spot size over a 24 micrometer interval. This resulted in two digital images that are 2,048 X 2,048 pixels. A pixel side corresponds roughly to four feet on the ground, but a mismatch or parallax error of one pixel results in about 7.7 feet of vertical error on the ground. The mensuration and interior orientation of the photos were also performed at ETL.

Figure 6-1 is image A of the stereo pair. The size of this digital image is approximately 1,260 pixels by 2,032 scan lines. It is considerably enlarged for illustrative purposes and has been digitally enhanced since the gray-scale range of the original digital data was rather narrow. The black lines on the picture are the evenly spaced matching grid lines in the parallax direction. These lines are spaced eight scan lines apart. Along these lines matching occurred at every tenth pixel, so for this image section, 53,910 match points were generated. A majority of these points were matched using a 21 X 21 pixel correlation patch over a seven-site search segment.

Figure 6-2 shows the conjugate grid superimposed on image B. This grid plot was produced by plotting straight lines between all the match points lying in the same column. The plot, then, is actually a picture of the match points and also of the parallax function that relates image B to image A. If Figure 6-1 and Figure 6-2 were viewed under a stereoscope, the corresponding lines would fuse stereoscopically and the grid would appear to lie on the terrain in three-dimensional space.

The match points were photogrammetrically intersected using the absolute orientation elements to produce a file of digital terrain data. The elevations were then contoured at 20-foot intervals. The resulting digital contour

Figure 6-1 Image A with Evenly Spaced Grid Superimposed

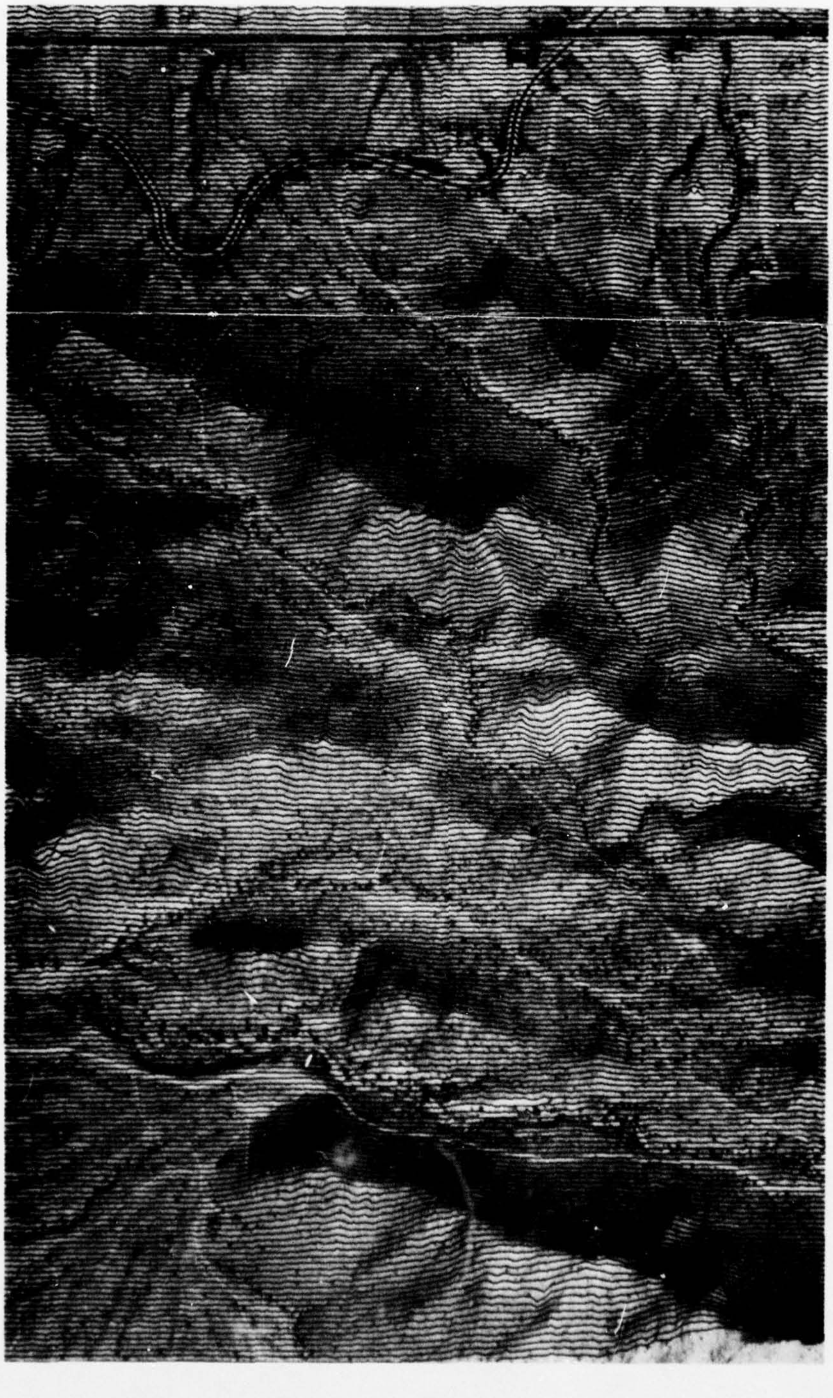


Figure 6-2 Image B with Conjugate Grid Superimposed

image superimposed on image A appears in Figure 6-3. Neither the terrain data nor the contour lines here have been smoothed. The contouring procedure makes use of local surfaces defined by bicubic polynomials that fit the data exactly; least squares techniques are not used. Since this contour image is generated in image A space, it is unrectified with respect to model or object space. However, the contour labels on the edges do represent actual feet above sea level on the ground.

Figure 6-4 is a plot of the reliability factor for each match point superimposed on a lightened version of image A. The more unreliable a match point is with respect to the five reliability criteria, the darker is its gray-scale value in this picture. For this processing example, of the 53,910 match points processed, 72% are reliable. Most of the unreliable areas observed in the picture are due to a low standard deviation of the image intensity in those areas. In these pictures, north is to the left. A large number of unreliable areas consistently fall on the north slopes of the mountain ranges. These slopes are highly illuminated and notably lacking in feature content. A specific example is the circular area at the top edge of the picture. Looking back at Figure 6-1, it is evident that this unreliable area lies on a steeply sloping, conical peak whose entire northern exposure is rather featureless. Matching was very difficult in this area, and the mismatching is manifested in the false contouring of this peak in Figure 6-3. However, the matching process did not break down completely after encountering this area. The match point reliability increased as the feature content improved on the other side of the peak. Other unreliable areas were not as dramatically inaccurate; in fact, some - though flagged as unreliable - were remarkably accurate.

Figure 6-3   Image A with Digital Contours Superimposed

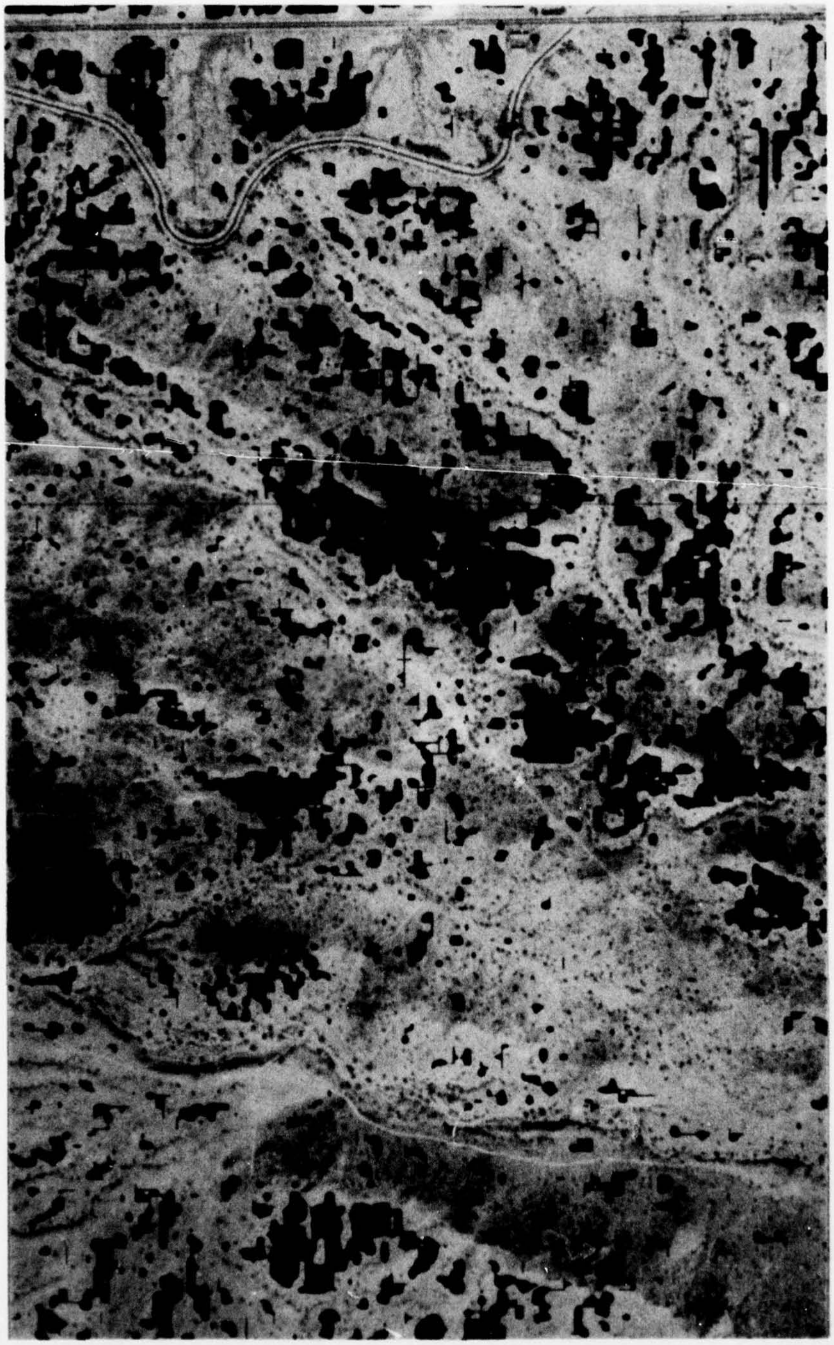


Figure 6-4 Image A with Reliability Plot Superimposed

7.0 CONCLUSION

All of the strategies that collectively make up the stereo matching algorithm described in this report have analytical justification and are theoretically predictable. However, in the face of real imagery of various kinds, there are so many variables involved relating sensor geometry, exposure conditions, and terrain and image characteristics that at times the algorithm appears to have a mind of its own. Algorithm tuning, at this stage, is rather an art than a science or engineering exercise. Trial and error procedures based on the accumulated experience of the user-analyst are required to obtain maximum algorithm performance.

If it is at all feasible or practical to tabularize and parameterize image, sensor, and terrain events, then the tuning of the algorithm to the conditions can be made more automatic and reside more in the realm of engineering than art. This is the subject of study in the next phase of the contract.

APPENDIX A

DIGITAL INTERIOR ORIENTATION: A PROCEDURE FOR REDUCING DIGITAL SCAN COORDINATES TO CALIBRATED PHOTO COORDINATES

DIGITAL INTERIOR ORIENTATION

A PROCEDURE FOR REDUCING DIGITAL SCAN COORDINATES TO CALIBRATED PHOTO COORDINATES

A procedure for reducing digital scan data coordinates to calibrated photo coordinates is necessary when photogrammetric computations are to be performed with digital image data.

Three distinct coordinate systems are to be considered:

- digital scan coordinate system
- reseau coordinate system
- fiducial coordinate system

Basic image manipulations are typically performed in the digital scan coordinate system; that is, in terms of scan lines and pixels. A reseau system is typically deposited on mapping images to remove film distortion. The fiducial system defines the optical axes of the mapping sensor and forms the base of all photogrammetric computation. The task then of digital interior orientation is to transform the digital scan coordinate system through the reseau system to the calibrated fiducial system. The effect of this transformation is to remove any film distortion that exists on the original film image and also to correct for scanner induced distortions.

Figure 1 describes the marks to be used in the transformation procedure. Fiducial marks A, B, C, and D are primary fiducials, whose intersection defines the principal point of the photo coordinate system. Fiducials E, F, G, and H are secondary fiducials.

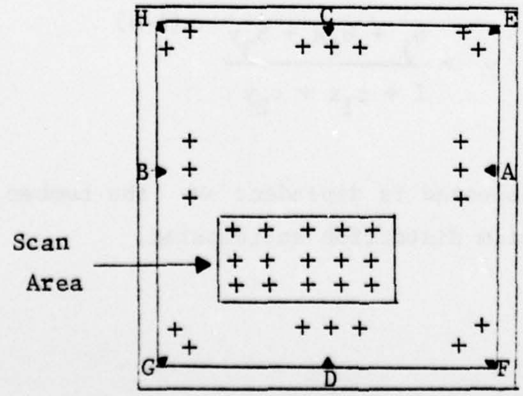


Figure 1

The calibrated coordinates of all of these fiducial marks are generally supplied in the calibration certificate for a particular mapping camera and lens. The shape of the fiducial marks and the reseau pattern varies with the type of camera. The reseau marks are generally etched on a glass plate in front of the film and deposited on the film at the time of exposure.

The calibration procedure for digital scan coordinates is as follows:

1. On the original film image measure a series of reseau marks for each local neighborhood around the fiducial marks.

   Using these measurements construct for each neighborhood a transformation T_R from the measured reseau marks to the calibrated reseau marks:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} T_R \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

where x',y' are the coordinates of the calibrated marks and x,y the coordinates of the measured marks.

This reseau transformation can take on one of the following forms:

$$\begin{aligned} x' &= a_1 + a_2x + a_3y \\ y' &= b_1 + b_2x + b_3y \end{aligned} \qquad (1\text{-}1)$$

$$\begin{aligned} x' &= a_1 + a_2x + a_3y + a_4xy \\ y' &= b_1 + b_2x + b_3y + b_4xy \end{aligned} \qquad (1\text{-}2)$$

$$\begin{aligned} x' &= a_1 + a_2x + a_3y \\ y' &= b_1 - a_3x + a_2y \end{aligned} \qquad (1\text{-}3)$$

$$x' = \frac{a_1 + a_2x + a_3y}{1 + c_1x + c_2y} \qquad y' = \frac{b_1 + b_2x + b_3y}{1 + c_1x + c_2y} \qquad (1\text{-}4)$$

The type of transformation selected is dependent on the number of marks measured and the nature of film distortion anticipated.

2. Measure the fiducial marks on the film and transform them to the calibrated reseau system using the previously derived transformations:

$$\begin{bmatrix} x'_F \\ y'_F \end{bmatrix} = \begin{bmatrix} T_R \end{bmatrix} \begin{bmatrix} x_F \\ y_F \end{bmatrix}$$

where x_F, y_F are the measured coordinates of a fiducial mark and x'_F, y'_F are the coordinates of the mark with respect to the calibrated reseau system.

This transformation removes the effect of film distortion on the individual fiducial marks.

3. Using the above-derived fiducial coordinates calibrated with respect to the reseau, next construct the transformation T_{RF} from the calibrated reseau system to the calibrated fiducial system:

$$\begin{bmatrix} x''_F \\ y''_F \end{bmatrix} = \begin{bmatrix} T_{RF} \end{bmatrix} \begin{bmatrix} x'_F \\ y'_F \end{bmatrix}$$

where x'_F, y'_F are the fiducial coordinates derived above and x''_F, y''_F are the calibrated fiducial coordinates supplied in the camera calibration certificate.

Again, the coefficients of the transformation T_{RF} can take the forms (1-1) to (1-4) mentioned above.

4. On the digital image corresponding to the scan area of Figure 1 measure a series of reseau marks and express the coordinates in the digital scan coordinate system.

Using these measurements, construct a transformation T_{SR} from the digital scan coordinate system to the calibrated reseau system:

$$\begin{bmatrix} x_{CR} \\ y_{CR} \end{bmatrix} = \begin{bmatrix} T_{SR} \end{bmatrix} \begin{bmatrix} I \\ J \end{bmatrix}$$

where I,J are the measured digital coordinates of the selected reseau marks and x_{CR}, y_{CR} are the corresponding coordinates of the calibrated reseau marks.

Again, the transformation T_{SR} can take the forms (1-1) to (1-4).

The effect of this transformation is to remove both the original film distortion local to the desired scan area and any geometric distortion introduced in the scanning and digitization process.

5. Construct the composite transformation T_{SF} from the digital scan coordinate system to the calibrated fiducial system:

$$\begin{bmatrix} T_{SF} \end{bmatrix} = \begin{bmatrix} T_{RF} \end{bmatrix} \boxdot \begin{bmatrix} T_{SR} \end{bmatrix}$$

where $\boxdot$ is a transformation composition operation (typically a matrix multiply).

The overall result of this calibration procedure is that any pixel (or point) whose coordinates are I,J in the digital scan coordinate system can be related to the calibrated photo coordinate system for subsequent photogrammetric computations. This relation is:

$$\begin{bmatrix} x_P \\ y_P \end{bmatrix} = \begin{bmatrix} T_{SF} \end{bmatrix} \begin{bmatrix} I_P \\ J_P \end{bmatrix}$$

where x_P, y_P are the calibrated photo coordinates of point P.

If a particular photogrammetric application requires the transformation from photo coordinates to digital scan coordinates, then the operation may be inverted, such that

$$\begin{bmatrix} I_P \\ J_P \end{bmatrix} = \begin{bmatrix} T_{FS} \end{bmatrix} \begin{bmatrix} x_P \\ y_P \end{bmatrix}$$

where

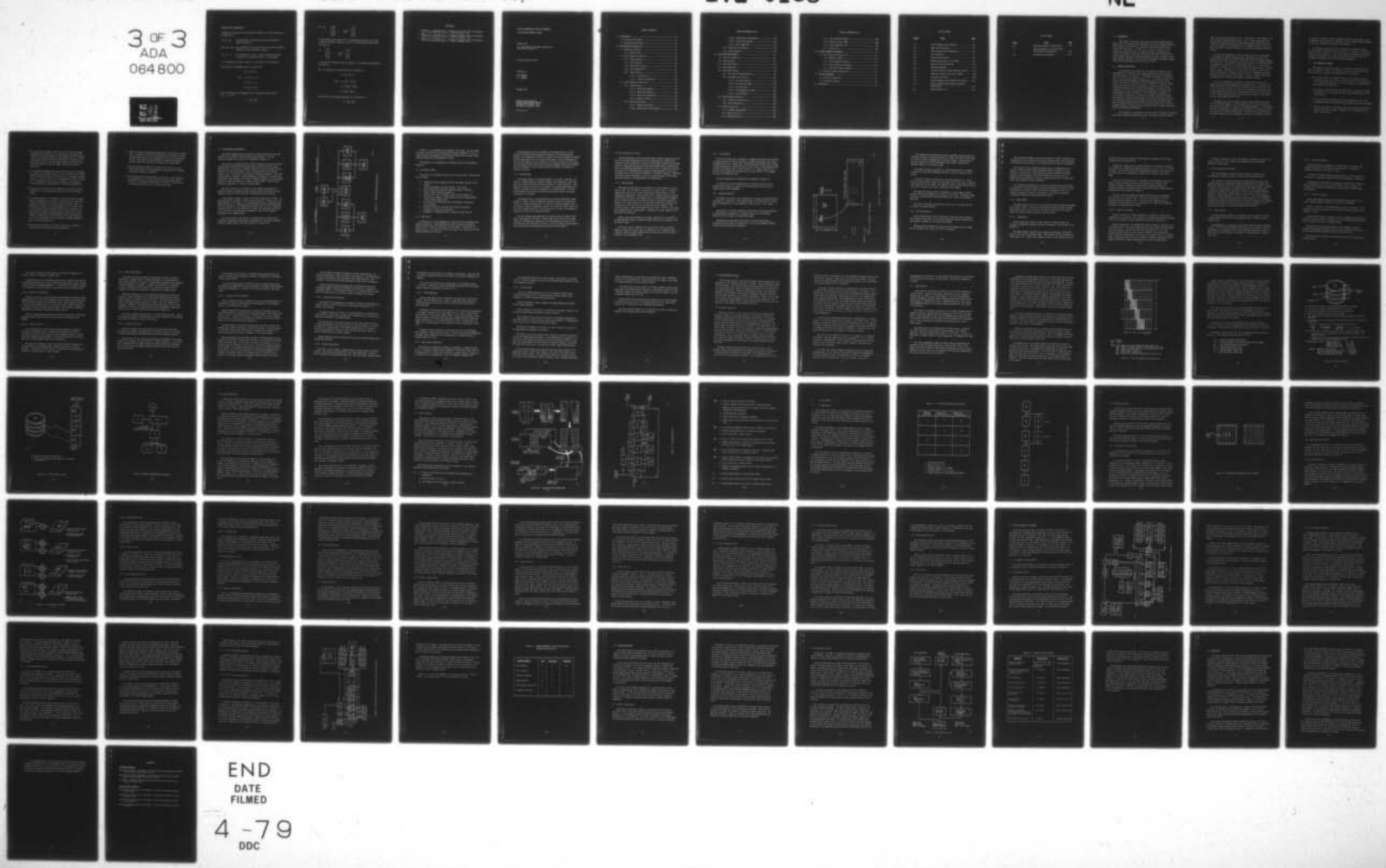
$$\begin{bmatrix} T_{FS} \end{bmatrix} = \begin{bmatrix} T_{SF} \end{bmatrix}^{-1}$$

APPENDIX B

EPIPOLAR LINE DETERMINATION

AD-A064 800   CONTROL DATA CORP MINNEAPOLIS MINN DIGITAL IMAGE SYS--ETC F/G 8/2
DIGITAL CARTOGRAPHIC STUDY AND BENCHMARK.(U)
DEC 78 D J PANTON, M E MURPHY, D S HANSON   DAAG53-75-C-0195
UNCLASSIFIED   CDC-76003-78-R6   ETL-0168   NL

3 OF 3
ADA
064800



END
DATE
FILMED
4-79
DDC

EPIPOLAR LINE DETERMINATION

Consider the following relative orientation elements that relate exposure #1 to exposure #2.

(B_x, B_y, B_z), the coordinates of exposure station #2 with respect to exposure station #1.

$(W_{21}, \emptyset_{21}, K_{21})$, the coordinate rotation angles that relate photo coordinate system #2 to photo coordinate system #1.

$(x_1, y_1, -f)$, the coordinates of a point in photo coordinate system #1 through which an epipolar plane is to be determined.

It is assumed that the focal length, f, is the same for both exposures.

The equation of the epipolar plane in system #1 is:

$$Ax + By + Cz = 0$$

where

$$A = -(f\,B_y + y_1\,B_z)$$

$$B = f\,B_x + x_1\,B_z$$

$$C = y_1\,B_x - x_1\,B_y$$

Then, the equation of the epipolar line in system #1 passing through $(x_1, y_1, -f)$ is:

$$y = -\frac{A}{B}x + \frac{C}{B}f$$

Now, let $\begin{bmatrix} B_x^* \\ B_y^* \\ B_z^* \end{bmatrix} = \begin{bmatrix} A_{21} \end{bmatrix}^T \begin{bmatrix} B_x \\ B_y \\ B_z \end{bmatrix}$

be the baseline vector expressed in a coordinate system parallel to photo coordinate system #2, where A_{21} is the 3 X 3 rotation matrix derived from W_{21}, $\emptyset_{21}$, K_{21}.

Let $\begin{bmatrix} x_1^* \\ y_1^* \\ z_1^* \end{bmatrix} = \begin{bmatrix} A_{21} \end{bmatrix}^T \begin{bmatrix} x_1 \\ y_1 \\ -f \end{bmatrix}$

be the point of interest system #1 expressed in a coordinate system parallel to system #2.

Then, the equation of the epipolar plane in system #2 is:

$$Dx + Ey + Fz = 0$$

where $D = z_1^* B_y^* - y_1^* B_z^*$

$$E = -(z_1^* B_1^* - x_1^* B_z^*)$$

$$F = y_1^* B_x^* - x_1^* B_y^*$$

The equation of the conjugate epipolar line in system #2 is:

$$y = -\frac{D}{E} x + \frac{F}{E} f$$

REFERENCES

1. Panton, D. J. and Murphy, M. E., "Digital Cartographic Study and Benchmark: First Interim Technical Report," to USAETL, October, 1975.
2. Panton, D. J. and Murphy, M. E., "Digital Cartographic Study and Benchmark: Second Interim Technical Report," to USAETL, December, 1975.
3. Panton, D. J. and Murphy, M. E., "Digital Cartographic Study and Benchmark: Third Interim Technical Report," to USAETL, September, 1976.

DIGITAL CARTOGRAPHIC STUDY AND BENCHMARK

FIFTH INTERIM TECHNICAL REPORT

Prepared for:

U.S. Army Engineer Topographic Laboratories
Fort Belvoir, Virginia

Contract DAAG53-75-C-0195

Prepared by:

D. S. Hanson
M. E. Murphy
D. J. Panton

December 1978

Control Data Corporation
DIGITAL IMAGE SYSTEMS DIVISION
2800 East Old Shakopee Road
Minneapolis, Minnesota 55420

76003-78-R5

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont.)

TABLE OF CONTENTS (Cont.)

LIST OF FIGURES

LIST OF TABLES

1.0 INTRODUCTION

This report is the fifth in a series of Interim Technical Reports which cover the development and implementation of stereo mapping algorithms and the design of an interactive processing system that can be used for terrain data collection. In particular, the results of Phase E of the program are contained herein. The primary purpose of this Phase was to initiate the design of a parallel processing system which incorporates the digital mapping algorithms developed previously and which includes an interactive editing capability. This system has been denoted the Digital Mapping System and will henceforth be referred to as the DMS.

1.1 UNDERLYING PRINCIPLES

The benchmark test of Phase C of this effort has shown that the implementation of production stereo mapping algorithms on general purpose computing machinery is impractical. Flexibility is inherent in these systems, but the required production speed is not. To satisfy both the flexibility and speed requirements, a parallel processing architecture was selected as a practical alternative for the benchmark test and for the DMS design. The parallel network incorporates a collection of fast, microprogrammable processors that function cooperatively, in parallel, in the performance of a task. The approach used to implement an algorithm on a configuration of these processors is one of distributive computing. In this approach an algorithm is divided into its basic functional modules. These modules are then distributed among the available processors such that each processor performs its given section of the algorithm simultaneously and in a pipeline fashion with the other processors and sections. The independent nature of the processors and algorithm modules provides flexibility and the parallelism provides speed.

The flexibility is necessary so that the system can respond to changing requirements regarding imaging sensor type, image scale, algorithm refine-

ments, and output data characteristics. A side benefit of the modularity is that the DMS can become multifunctional. That is, when the system is not being used for production stereo compilation, the basic system modules can be used individually as development tools for change detection, projective model development, image rectification, or for general purpose scanning and display.

But speed and flexibility are not the only criteria to be considered in the design of a digital mapping system. Extremely fast systems that automatically compile terrain data already exist within the mapping community. The limitation is that automatic systems cannot fulfill the expected accuracy requirements all the time, for every possible image type or terrain event. Moreover, an automatic system can hardly match the sophistication of a human stereo compiler. Therefore, in current operational practice, much time is spent as a follow-up to automatic compilation, in an independent operation involving data correction and data fill-in. This is a manual process that consumes much more time than the automatic compilation process.

It is not unreasonable to consider the automatic functions and the human functions as being integrable into a single system. In this way, the man and the machine can cooperatively perform the entire terrain data collection task in one session. The computer is assigned the operations that it does best and the man performs the functions for which he is most suited. All data resides within the same centralized system and there is no need to set up a stereo model more than once.

Throughput then, for an integrated system, must be measured in terms of total system throughput, from the mounting of the film on the scanner to an acceptable end product, not just the throughput capability of the computer. The unit of measurement for comparing systems should be the total wall clock time required per complete stereo model. It appears from the design effort initiated in Phase E that an integrated system, such as the DMS, with a man

in the loop, can achieve a total throughput rate that is much greater than the methods currently employed which use very fast automatic devices followed by very slow, independent, manual fill-in processes.

An additional design factor is the capability of the DMS for growth. The basic modular construction allows processors to be added to the system to enhance machine performance. Also, entire channels of processors and display stations may be added to the system to provide more parallelism both for throughput enhancement and for multifunctional capability.

1.2 DMS OPERATIONAL SCENARIO

The following step-by-step scenario is provided to illustrate how the DMS is designed to perform. The individual steps below represent the major functions that are performed in different parts of the system.

- The operator mounts the panoramic materials to be digitized on the film reader, which can be part of an asynchronous, off-line subsystem to the compilation subsystem. The film is aligned to be consistent with the desired stereo model area.

- The films are scanned and the resulting data is compressed using DPCM. The digital data is organized into block format and deposited on the system disks.

- Exterior orientation parameters and point measurements from a prior triangulation process are entered by the operator into the host computer.

- Interior orientation is performed by the system after the operator has assisted the system in making a conjugate set of measurements on the digital image display.

- The rectification processor takes over and rectifies both Image A and Image B of the stereo pair. Rectification is performed with respect to the stereo airbase such that epipolar lines are parallel to one another and also parallel to the digital scan raster. The rectification process uses the blocked image data from the system disks and outputs the rectified imagery back onto the disks in block form.

- As rectification proceeds, the operator has the option of reviewing the imagery on the display device. He can, with his cursor, define adverse areas which could prove difficult for the automatic matching process. A review at this time is also important so that the operator can familiarize himself with the imagery in an attempt to better estimate the tuning parameters for the matching algorithm.

- The operator enters into the host computer the appropriate tuning parameters for the matching algorithm and initiates the matching process.

- As matching progresses, the imagery is scrolled through the display window in anaglyphic stereo fashion. Match points are connected by straight line segments to provide match point profiles which are graphically overlayed on the stereo display. When it appears that the matching process is encountering difficulty, the operator can stop the process, manually insert or correct match points with his stereo cursor, change the tuning parameters, back up the process, restart the matching, or all of the above. He can also flag areas for later, more extensive editing and obtain a hardcopy snapshot on paper of the questionable area.

- When matching is complete, the operator can request a rematch of any desired area with different tuning parameters.

- After all automatic matching and rematching, the edit process is initiated. By means of the various editing modes, the operator can clean up erroneous data and fill in missing data directly from the display station. The system responds by making the appropriate changes to the match point file that has been created on the system disks.

- When all editing is complete, the match point file is sent to the intersection processor for photogrammetric intersection. No further operator intervention is necessary.

- The resulting uneven and over-dense terrain grid is then normalized by interpolation into the final product with the required regular spacing. The resultant normalized grid is written to magnetic tape to complete the processing for a single stereo model.

2.0 DMS FUNCTIONAL DESCRIPTION

The Digital Mapping System is intended to satisfy present and future DMA requirements for Digital Terrain Elevation Grid production. The major functional elements of the system are illustrated in Figure 2-1.

The system is an automatic digital stereo compilation device with process monitoring and interactive editing capabilities sufficient to yield output products that are internally as complete, accurate, and self-consistent as the quality of the input data permits. The coordinate system of the compiled elevation data is determined by the exterior orientation parameters. The final output product is expressed in three-dimensional coordinates on a regular grid conforming to DMA standards. The system is upwardly compatible with an expanded system to include the capability to adjust adjacent models to match without discernible boundaries in a common coordinate system.

The input data used by the system is stereo imagery supplemented by exterior orientation parameters computed during off-line mensuration and triangulation, operations which are part of existing production processes.

Stereo imagery submitted as input data for the system may be on film. An image digitizer is provided to convert the film products to digital form. The image digitizer is one element of a subsystem which prepares a complete stereo model for compilation. The hardware and software used in the data preparation subsystem perform at a rate sufficient to support the simultaneous operation of at least three stereo compilation subsystems. Each of these operates independently on a separate model.

The output product of the system is a digital terrain elevation grid, conforming to DMA product specifications appropriate for the mission although limited in spatial extent to the area covered by a single model.

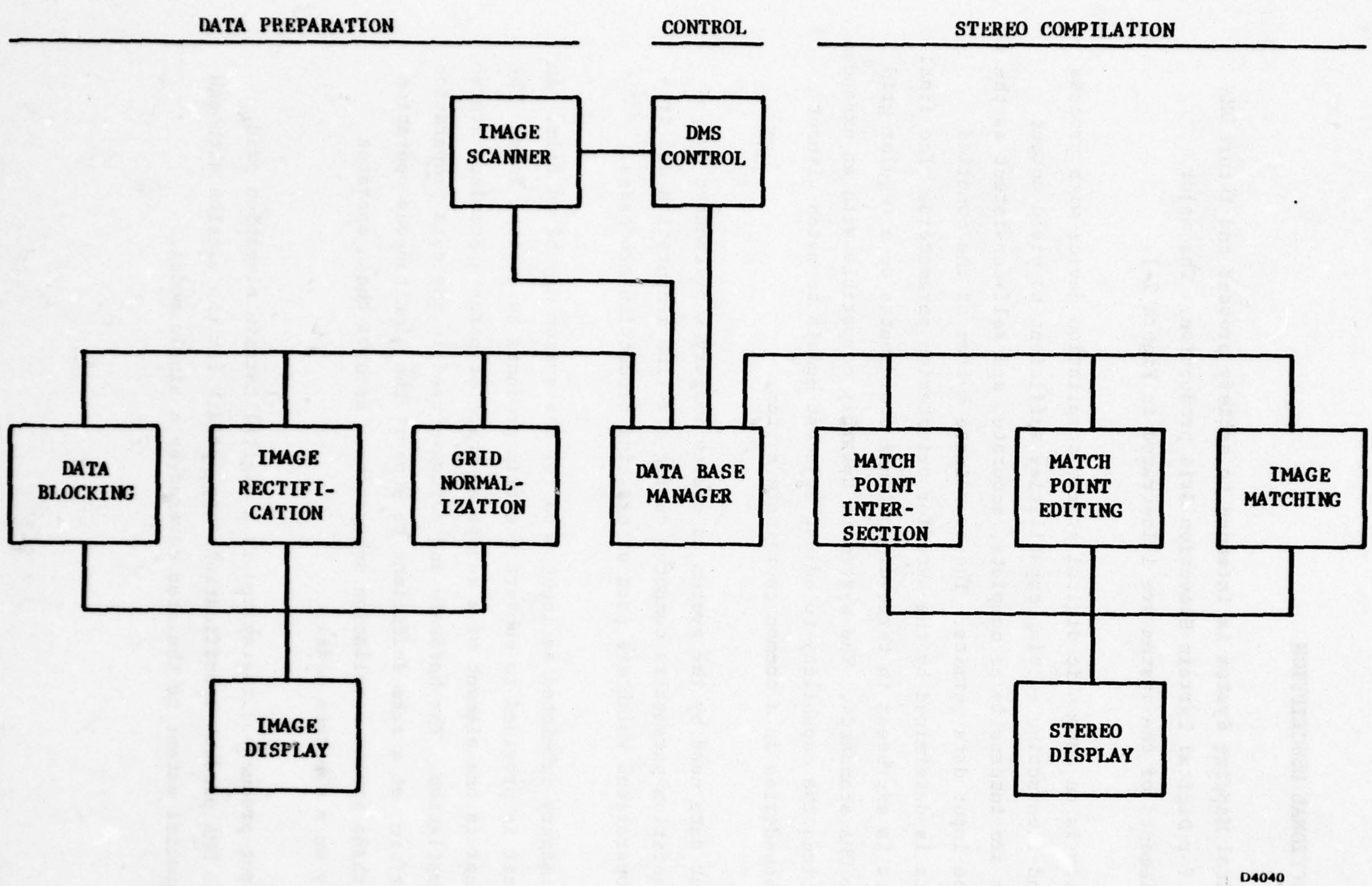


Figure 2-1 Digital Mapping System Functions

Figure 2-1 is a functional block diagram of the system. The two primary functions of the DMS are Data Preparation and Stereo Compilation. Both of these functions are supported by a Host/Control Module which includes a host computer and DMS data base management software.

The functions to be implemented on the DMS are given in the diagram of Figure 2-1.

2.1 Host/Control Module

This module is the primary executive of the entire system. Its functions are as follows:

- Communicate with the operator and the environment external to the system.
- Control and sequence the other modules of the system.
- Run the manufacturer-supplied operating system to provide a general purpose computing capability.
- Supervise the input of parametric data for each image of the model and the staging of input image data from the image scanner or high density magnetic tape.
- Perform interior orientation and transformation construction for each image.
- Perform the exterior orientation and relative orientation necessary for epipolar rectification.
- Management of the DMS Data Base to support system functions.

2.1.1 DMS Control

The DMS Control can be implemented as a typical 16-bit minicomputer with standard peripherals. These peripherals include a keyboard and character display (CRT) for operator interaction with the system, a card reader for software input, a line printer for hard copy output, and a system disk for the operating system and file system.

The performance of the host computer is not speed critical, for the production functions are performed by the network of Flexible Processors. Moreover, the computational load that is placed on the host computer is extended over the processing time of the entire stereo model. The only requirement is that the host computer form an interface to the other processors in the system. I/O data channels such as the Unibus in Digital Equipment Corporation equipment and the A/Q channel in Control Data Corporation equipment are examples of suitable interfaces.

2.1.2 DMS Data Base

The primary purpose of the DMS Data Base is to provide a temporary, non-archival, storage facility for digital imagery, model parameters, match point files, and terrain elevation data which pertains to current production. Data handling is required to support the operation of the primary DMS functions at rates not limited by access to information in the DMS Data Base. The Host/Control Module must provide data for the process memory used by the primary functions as it is needed.

The hardware units of the DMS Data Base must provide sufficient storage capacity to permit the Data Preparation function and the Stereo Compilation function to operate simultaneously. In addition, sufficient capacity should be allowed to contain intermediate products of at least one inactive model for each of the primary functions. This allows work on a model to be temporarily suspended in favor of more urgent priorities.

The data storage medium employed by the DMS Data Base is a bank of high speed, large capacity disk units. Digital image source data from the image scanner is compressed and reblocked by the Data Preparation functional area and directed to the Data Base for storage. Control information and mensuration data are entered into the Data Base via the Host Control computer.

2.2 Data Preparation Functions

The Data Preparation function includes tasks related to preparing the raw data for the Stereo Compilation function and adjusting the elevation grid produced by Stereo Compilation to meet the final product specification for DMS. Except for Image Display, operations within Data Preparation are exclusive and may not be performed simultaneously in the same system. Sequential operations are chosen over simultaneous ones because no single operation is very time consuming and the total hardware requirement is otherwise increased. Operations included in the Data Preparation function are Image Scanning, Data Blocking, Image Rectification, Grid Normalization and Image Display.

2.2.1 Image Scanning

The Image Scanning operation is performed to convert imagery on film to digital form. The film image is represented in digital form as a sequence of numbers which describe the relative monochromatic image density of a film transparency sampled with a fixed aperture size at regular intervals over the useful area of the image. Each sample point in a two-dimensional rectangular array of sample points over the useful area must be precisely located with respect to all other sample points in the array and accurately measured, and the position of the array must be accurately known with respect to film calibration marks located some distance from the array. The size of the image area scannable must be large enough to accommodate all source imagery used in DMA production operations.

Digital image data produced by the image scanning task is introduced to the DMS Data Base through a transient storage buffer under supervision of the DMS Control Module.

The digital data is organized into records which consist of adjacent samples of image density which span one dimension of the rectangular array. A sufficient number of equally spaced records is provided to span the second dimension of the rectangular array.

2.2.2 Data Blocking

The Data Blocking task is performed to reorganize the digital data produced by image scanning into records consisting of samples distributed over a small two-dimensional block of image area. The DMS Data Base provides buffer storage of image scanning data. The Data Blocking task consists of selecting data from records in the buffer memory to span a distance of n samples along each of m records, and providing the selected data as a single record of n x m samples for entry in the DMS data base.

The data organization is schematically illustrated in Figure 2-2.

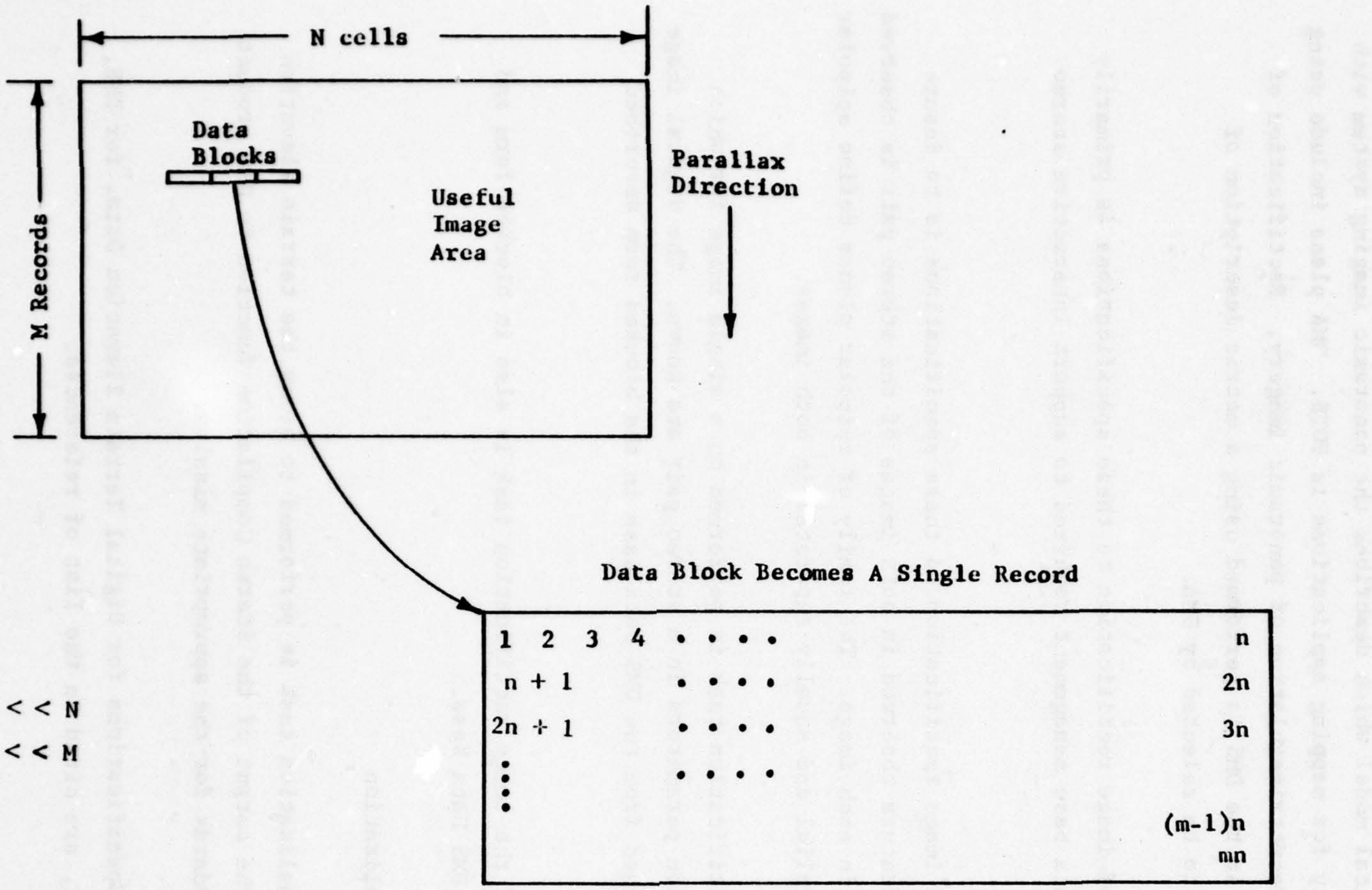
The purpose of the Data Blocking task is to minimize the size of the process memory required to support the Stereo Compilation function and to simplify the data base management.

2.2.3 Image Rectification

The Image Rectification task is performed to reorient the digital data for each image of a stereo pair. The rectified image is obtained from the initial image by reprojection to a plane parallel to the air base of the stereo pair.

Rectification of central perspective stereo pairs to these specifications is accomplished by implementing in a digital algorithm, the operations performed on a film image by a rectifying enlarger.

Rectification of panoramic stereo pairs to these specifications is accomplished by performing a rectification as for central perspective images, although involving additional parameters.



Example: $n = 256$

$m = 8$

Block Size = Record Size = 2048 Samples

D4306

Figure 2-2. Image Data Organization

The mathematical model which describes the panoramic imaging system with sufficient accuracy for mapping applications is POTS. DMA plans include using this model for direct triangulation of panoramic imagery. Rectification of panoramic imagery in the DMS is performed using a metric description of panoramic imagery to be selected by DMA.

The purpose of image rectification to these specifications is primarily tc simplify the data base management required to support interactive stereo display.

The result of image rectification to these specifications is to insure that any terrain feature observed in both images of the stereo pair is observed at the same scale in each image. The family of epipolar planes define epipolar lines that are parallel and equally separated in both images.

The image rectification task is performed on a single image for which relative orientation parameters in a stereo pair are known. The digital image data used is obtained from the DMS Data Base in the blocked form described previously.

The output of the image rectification task is also in blocked form and is entered in the DMS Data Base.

2.2.4 Grid Normalization

The Grid Normalization task is performed to bring the terrain elevation grid obtained as the output of the Stereo Compilation function to DMA product specification standards for the appropriate mission.

DMA Product Specifications for Digital Terrain Elevation Data, for DMS, and for Firefinder, are cited in the list of references.

The operations performed include the generation of header information and supplementary data as required by the product specifications, and interpolation to the regular grid specified by the appropriate mission product specification.

The accuracy of the interpolated output product is determined by the density of points from the Stereo Compilation function compared to those produced by interpolation. DMA operational practices determine the actual density of points produced by the Stereo Compilation function; the Grid Normalization task operates to produce an output grid with more or with fewer points than provided by Stereo Compilation.

The interpolation method used to produce the normalized grid is selected by the operator from the repertoire of options provided in the system. Linear interpolation is provided in the initial system configuration and is the default option when no selection is specified. The system is designed to accommodate software for other interpolation methods.

2.2.5 Image Display

The Image Display task is a set of interactive display operations performed under operator control to verify the Image Scanning, Data Blocking, and Image Rectification results and to perform an interior orientation of the digital image in the image coordinate system of the source data.

2.2.5.1 Verification

Digital image data created as output product of Image Scanning, Data Blocking, and Image Rectification tasks may be examined by the operator at his discretion.

The Image Scanning task data may be sampled by displaying a single CRT screen of data from the digitized image. The displayed data may be sequential image samples, i.e., pixels, from sequential records. This display scrolls by

removing the top line, shifting all lines upward, and adding a new line from the most recently scanned record.

Display of a larger area at a reduced resolution is obtained in a similar way by choosing to display the average pixel from a 2 x 2, 3 x 3, or 4 x 4 block of pixels. In any case the operator may change the position in the record of the first pixel to be displayed; the display can be scrolled to the left or right.

Vertical scrolling continues at a constant rate controlled by the rate at which the scanner completes records until stopped by the operator. When vertical scrolling is stopped the display may be examined carefully but scanning continues. Data scanned while vertical scrolling is stopped cannot be displayed until Data Blocking is completed.

The data from Data Blocking and Image Rectification may be displayed at full resolution or at reduced scale. Scrolling in horizontal and vertical directions is under operator control. Data from these operations can be displayed only after completion of an operation and is taken from the DMS Data Base where the completed results reside.

2.2.5.2 Interior Orientation

Interior orientation of digital image data is required to permit metric computations to be performed. The operation is performed after the Data Blocking task as a pre-requisite to Image Rectification and Stereo Compilation.

Interior Orientation is accomplished by using an operator controlled cursor to identify points in the digital image which have known image coordinate values from triangulation. The scan line and pixel count specified by the cursor position are paired with the known x and y coordinates of the image point (which may be photo-identifiable, a pug point, and a reseau point, or a fiducial mark.) Several such pairs are used to specify a polynomial transformation from digital coordinates to image coordinates and the inverse transformation.

The actual computation of the transformation is performed as part of the Control function by a host computer upon completion of the interactive gathering of point pairs.

2.3 Stereo Compilation Functions

The Stereo Compilation function includes operations related to the determination of terrain elevation at points within the stereo model.

The operations of Stereo Compilation are Image Matching, which is the automatic process for identifying conjugate image points of the stereo pair, Match Point Editing, which is the interactive process which allows the operator to assist in the definition of conjugate points when adverse regions are encountered, and Match Point Intersection, which is the completely automatic process of intersecting conjugate points in model space to define terrain elevation. Model coordinates are normally chosen to coincide with geographic coordinates. Match point intersection is also performed on a set of manually selected points found in adjacent models to verify the integrity of the digital model setup with the block of models adjusted during triangulation. This verification precedes all other Stereo Compilation functions.

2.3.1 Image Matching

The Image Matching operation is performed to define a network of stereo conjugate match points which can be intersected later to define a terrain elevation grid.

Image Matching is an automatic process which requires operator assistance to begin, is monitored and evaluated continuously by the operator, and may be interrupted by the operator at any time and restarted at any point before or after the point of interruption. The product of Image Matching is a match point file.

2.3.1.1 Match Point Density

The match points are arranged on a regular array in one image. The conjugate array in the other image of the stereo pair is distorted by differential parallax of the terrain relief.

The network of terrain elevation points produced by the intersection of match points is irregularly spaced in horizontal orthographic coordinates.

The spacing of match points in one image is selected by the operator to yield reliable results consistent with the image scale, the density of elevation points required to describe the terrain relief, and DMA product specifications.

2.3.1.2 Match Point Display

As the Image Matching operation is performed, a stereo display of most recently matched imagery is produced for review by the operator.

Operator evaluation of the image matching quality is achieved by direct comparison of the surface defined by the terrain and the match points, both observed in stereo.

The match point surface is viewed in stereo, superimposed on the terrain surface by coloring pixels at conjugate point pair positions appropriately for stereo perception as an array of floating marks.

The quality of image matching achieved by the automatic process is monitored by observing the departure of the surface defined by the array of floating marks from the perceived surface of the stereo model.

The stereo display scrolls in the parallax direccion as the match point process proceeds.

The scale of the stereo display monitor is reduced by averaging 3 x 3 blocks of pixels for display as a single pixel.

The scrolling rate of the reduced scale display is limited by the speed of the matching process which in turn is affected by image scale, quality, and the operator selected processing parameters. A nominal rate for reference purposes is one minute for a terrain feature to pass across the stereo display screen.

2.3.1.3 Adverse Area Definition

The match point file contains adverse area flags which are inserted during Image Matching in two ways. The automatic process evaluates the reliability of each match point entered into the match point file. This reliability measure is used to define adverse areas; i.e. areas within which operator support may be required to define satisfactory match points. The reliability conditions which generate an adverse area flag are operator defined.

During the image matching process the operator has the option of defining additional adverse areas to receive further attention during Match Point Editing.

2.3.1.4 Operator Control

Active participation by the operator in Image Matching is required to initiate the automatic process, to interrupt the automatic process when observation of the match point display reveals inadequate results, to select new process parameters when necessary, to restart the process at the desired point, and to define adverse areas for later attention.

The operator is expected to monitor the stereo display to evaluate the results obtained on a continuing basis, taking active control only when necessary. Intervention may take the form of interruption and restarting, or definition of adverse areas for later attention.

2.3.2 Match Point Editing

The Match Point Editing operation is performed to define or redefine match points in adverse areas, to permit reexamination and redefinition of match points anywhere in the model, to permit manual compilation and comparison of results anywhere in the model, to permit the definition of an arbitrary surface for impossible adverse areas, and to define the useful portion of available third images for definition of match points in adverse areas.

Match Point Editing is a set of interactive processes which permit the operator to create a complete file of match points representing the actual terrain surface as accurately as the data allows. The interactive processes within the Editing operation make use of operator skill and judgement for selection of methods, for making difficult stereo perceptions, and for final assessment of results. The system software and data management capability is used to carry out operator decisions and requests automatically.

The product of Match Point Editing is an edited match point file. Adverse area flags which remain in the match point file after completion of the Editing task require another stereo model for resolution.

2.3.2.1 Adverse Area Review

Adverse area flags in the match point file are inserted during Image Matching by the operator upon perception of a difficult area or by the system to identify areas with a consistently poor match point reliability measure.

Adverse areas of either type may be reviewed by the operator. The review process may be complete by type (operator or system identified), or by specific area. Specific areas are tagged by type and sequence number upon definition. The type or the type and sequence number can be used to recall the area for display and review.

The operator has the option of scrolling through the entire model for review of the match points selected in addition to calling for the automatic review of match points.

At any time during manual or automatic review of selected match points the operator may suspend the review to reinitiate the automatic image matching task with new parameters or to invoke one of the following match point edit modes.

2.3.2.2 Adverse Area Profiling Mode

Stereo compilation within an adverse area can be accomplished manually if the operator is unable to adjust the processing parameters of the automatic process to yield satisfactory results.

Manual compilation is accomplished by holding the floating mark on the model surface while the system moves the mark horizontally along a model cross-section normal to the parallax direction. Operator control of the floating mark elevation is similar to that required by orthophoto projection plotters which use a slit for differential rectification.

Profiling begins on the profile intersecting the adverse area with the minimum value of model coordinate X or the model boundary. The floating mark is stationed on this profile just outside the adverse area boundary on the side with minimum value of model coordinate Y or the model boundary.

When the boundary is wholly within the model, profiling begins with the vertical position of the floating mark controlled by the system until the mark enters the adverse area. At this point the mark is constrained to lie in a vertical plane and continues to move in the Y direction at a speed controlled by the operator. The operator simultaneously controls the vertical position of the mark to lie on the model surface or on the surface estimated by the operator. The vertical gain of the elevation control is selected by the operator to comfortably perceive the apparent vertical motion of the mark.

Upon reaching the opposite boundary (of model or adverse area), the operator may approve or reject the profile just completed. The floating mark is moved by the system to the beginning of the profile indicated. The sequence of events continues until profiling the adverse area is completed.

During profiling within the adverse area, existing match point file entries are tagged as invalid and provided with pointers to a sequence of replacement entries. The replacement values lie on a regular array of points that is consistent with the match point density elsewhere in the model.

2.3.2.3 Adverse Area Fitting Mode

The adverse area fitting mode is provided as a method for producing some terrain elevation results within areas in which stereo perception is inadequate for profiling.

A polynomial surface is fitted to a qualifying number of points defined by the operator. The surface is evaluated at a regular array of points in one image that corresponds to the matching grid.

The floating mark of the stereo display is used, under operator control, to define the boundary of the area within which a fitted surface is to be used for definition of match points. The model boundary should not be used as a boundary along which the fitting is controlled unless the model surface is actually perceived along the model boundary.

Defined positions within the adverse area are used to provide higher order fitting when necessary.

2.3.2.4 Profile Editing Mode

This mode is used to repair a single profile or a small number of adjacent profiles. The conditions which generate local image mismatching, for which profile editing is the appropriate repair tool, are reseau marks, moving vehicles,

and blemishes which may appear in one image of a stereo pair. When this mode is invoked the operator defines the end points of the profile segments to be replaced.

The system then positions the floating mark on an acceptable portion of a profile. Profiling to generate replacement match points for the faulty segment proceeds as described in section 2.3.2.2.

2.3.2.5 Third Image Mode

Adverse areas which are not resolvable in one model may be resolved in another. When the DMS Data Base contains a third image which produces an overlapping model, the system can be used to evaluate the utility of the additional model.

Coordinates of the adverse area boundary in the current model are projected to the image coordinates of the third image and, if interior orientation has been performed, to the digital address range. If the required portion of the image is within the data base, the image can be flagged for later review and possible compilation. If the required portion is not in the data base, the image can be reviewed manually and evaluated for possible scanning at a later time.

Subsequent stereo compilation of an adverse area from any overlapping model produces a separate match point file. Results are processed independently and merged after completion of Grid Normalization in a manner similar to adjacent models in a block.

2.3.3 Match Point Intersection

The Match Point Intersection operation is performed to compute an array of X, Y, Z terrain coordinates which represents a sufficiently dense sample of terrain elevations to define a grid from which a regular network of final X, Y, Z coordinates can be interpolated. This is a completely automatic task.

The edited match point file is used as input. The product is a network of X, Y, Z terrain coordinates used by the Grid Normalization operation of the Data Preparation function.

2.3.4 Stereo Display

The Stereo Display operation is a set of interactive stereo display options involving the use of a floating cursor to permit operator stereo perception to override system defined match points.

The Stereo Display is used to support the Image Matching and the Match Point Editing operations.

Stereo perception of the model is provided by an anaglyphic display of the rectified digital images juxtaposed to clear Y parallax.

Stereo display of the match point surface is an anaglyph superimposed on the model surface in such a way that the match points appear as floating marks which coincide with or depart from the model surface.

The Match Point Display is the basic tool used to monitor the quality of the Image Matching performed by the system.

The operator controlled floating mark is used in an interactive process to initiate the Image Matching operation. As the floating mark is scanned in the Y direction by the system, the operator controls the vertical position of the mark. The system samples the mark position at periodic intervals to establish the image match points used to initiate the automatic image matching process.

The reliability measure associated with each match point selected by the system may be used to control the intensity or the size of the points on the match point surface display. The surface then appears as a network of points on the match point surface with unequal intensity or size. Extended areas with

points corresponding to a poor match point reliability become an immediate visual cue to the operator that local properties of the image or the terrain are causing difficulty for the automatic process.

The operator may use this visual cue to examine potential adverse areas more closely for possible manual correction. The system can make use of the reliability measure for automatic adverse area definition.whether or not the reliability display option is used.

The Match Point Editing operation also makes use of the stereo display. The operator may call for full scale display instead of the reduced scale display used for monitoring of the Image Matching operation.

Any stereo display requested by the operator may be with or without the display of the associated match point reliability.

3.0 DMS ALGORITHMIC MODULES

The following sections of the report describe the basic algorithms to be implemented in the software complement of the DMS. These are primarily computational packages. The communications and overhead modules are to be distributed around the system, and will be specified in detail at a later point in the system design cycle. Some of the modules that follow, such as stereo matching, image blocking, and analytical intersection, are well defined in terms of scope, complexity, and processor arrangement. Others, such as image rectification, display generation, and editing are more tentative at this time; their final design being contingent upon parameters and decisions to be supplied by supplementary image processing studies.

3.1 Interior Orientation

Interior orientation is the process of constructing transformations between the digital scan coordinate system of the imagery and the photo coordinate system defined by the calibration marks on the film. The need for such transformations arises from the fact that the DMS performs both digital image processing operations and photogrammetric computations. The digital image processing functions are performed in the image raster coordinate system, in terms of scan lines and pixels, whereas the photogrammetric functions are performed in the photo coordinate system. An additional need stems from the fact that the scanning of the imagery results in an image transformation. Thus, the digital image is a somewhat different geometric entity from the original film image. Interior orientation serves to model this image transformation by accounting for any scanner induced bias and distortion.

Interior orientation transformations are typically linear in form; however, higher order polynomials can be employed if non-linear film defects or scanner induced distortion are anticipated. The transformations are constructed by the least squares fitting of a set of measured points on the

digital imagery to a conjugate set of pre-calibrated or measured points on the film format. Typically, reseau marks are used for this purpose. But since there is no calibrated reseau on the panoramic materials, a well-distributed set of image feature points is sufficient.

In the DMS, interior orientation is performed by the operator using his interactive display station and by the host computer of the system. It is assumed that a collection of feature points has already been defined, marked, and measured on the panoramic film materials as part of the triangulation phase, separate from the DMS. These measurements are then treated as input to the DMS host computer. With the imagery already scanned and residing on the system disks, the operator locates one mark using his display and cursor to provide an approximate relationship between the display scan coordinates and the photo coordinates. The host computer then drives the display to the other feature points and the operator positions his cursor to make the final refined measurements.

A more automatic measurement approach can be implemented if the feature points on the film are marked using a standard, well-defined symbol. In this approach, the operator measures one point as described above, but thereafter the system visits the remaining points and employs a mark detecting algorithm to automatically determine the digital coordinates of the mark center. The system determination is then displayed for operator validation.

The result of this mensuration process is a conjugate set of measured coordinates. The host computer uses these coordinates to fit the interior orientation transformations. The fit residuals are displayed for operator approval, and in the case of unsatisfactory residuals, the mensuration process can be reinitiated.

As a final step, the host computer computes the interior orientation transformations with the exterior and relative orientation transformations that were supplied as input. The reason for this step is to eliminate successive point transformations later in the process. This allows

photogrammetric projections to be made directly from digital scan coordinates, thus increasing the efficiency of the matching, editing, and intersection processes.

3.2 Image Blocking

Image analysis using digital techniques is an attractive approach for several reasons. Among them are speed, repeatability, ease of operation, adaptability, and overall cost-effectiveness. One of the most important aspects of designing a digital image processing system is the provision for dealing with the enormous data volume. Consideration given to this aspect and recommended approach in the Digital Mapping System will be the subject of this section.

The design of the memory configuration in the Digital Mapping System is proposed as the best tradeoff given the variables of size, cost, and speed. There is sufficient memory to deal with the expected data volume; the cost has been kept reasonable; and the speed has been determined to be acceptable both in terms of production requirements and compatibility with other system modules.

The memory module in the DMS consists of large capacity, moderately fast disks, smaller and faster buffer staging memories, and a control processor. The disks provide large volume, low cost storage. To make the disks viable in terms of speed, a block organization is recommended for image data.

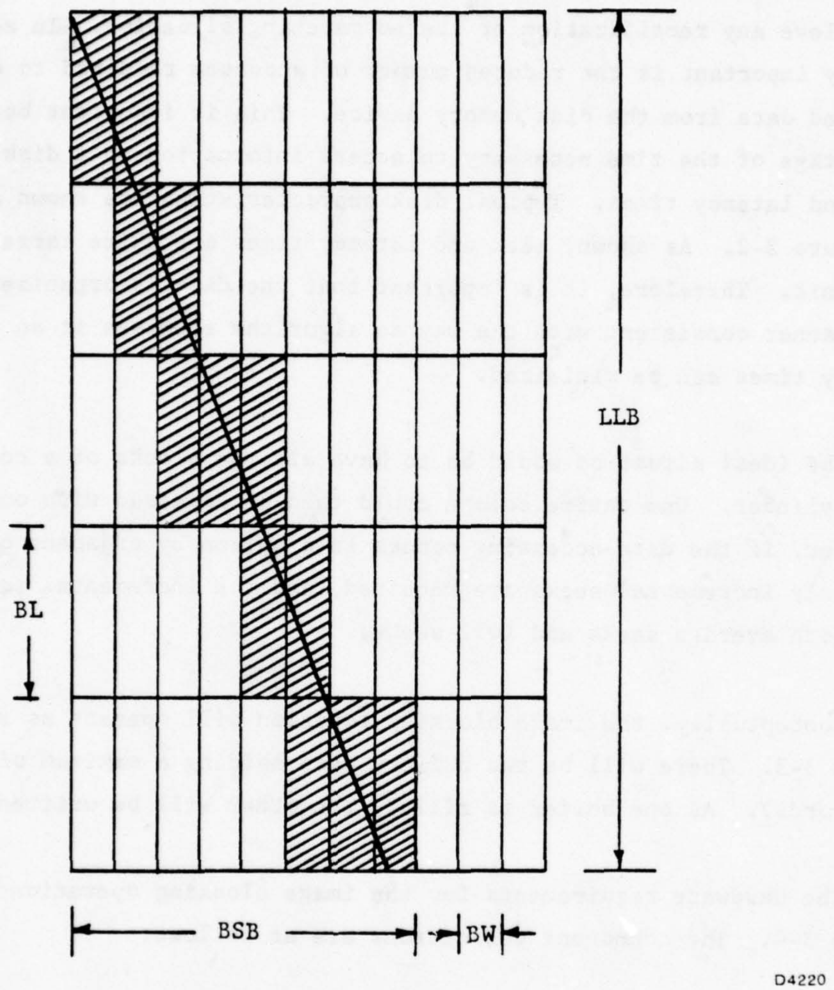
In a block organization scheme the digital image is partitioned and addressed in terms of blocks rather than in terms of scan lines. Block length and width are chosen in a way which will maximize the usable storage capacity of the disk subject to the constraint that efficient communication with other system modules must be possible.

The parametric relationships involved in the block organization are best explained with a discussion based on Figure 3-1. The diagram represents an input image section divided into blocks of length BL and width BW. An image scan line is oriented in the vertical direction of the diagram, such that BL is defined in terms of pixels and BW in terms of lines. The diagonal line represents the projection of an output line in input space; that is, the resampling of the input image occurs along that line to produce one line of output image. The diagram, therefore, depicts a typical image rectification situation, and is an example which illustrates image blocking. For stereo matching, a similar situation exists where the projection of a single scan line of Image A is not so linear, but crosses many scan lines of Image B. The matching situation is also a bit more complex, since the Image B resampling is performed on a patch basis. Nevertheless, basic block accessing concepts apply.

If the block accessing scheme were not employed, then the process memory required to resample along the diagonal line would have to be large enough to contain the number of complete scan lines equivalent to the BSB dimension of the figure. For severe cases of rectification or terrain relief displacement, a memory of this size is rather expensive.

As shown in the diagram, LLB is the number of blocks defined across the entire width of the image and is computed by dividing the number of pixels in a scan line by the block length in pixels. BSB is the maximum buffer size required for the rectification expressed in blocks and is defined as maximum buffer size in lines divided by block width.

The shaded blocks in the diagram represent the BRL blocks required to resample the given line. The process memory, then, need only be large enough to contain these blocks. This represents a 70% decrease in memory size compared to the non-blocked approach. In addition, once the memory has been loaded with BRL blocks on the first line to be resampled, it is necessary to update the memory with only NBRL new blocks for each succeeding line to be resampled. These new blocks replace the least recently used blocks already in the memory.



BRL = LLB+BSB-1
NBRL = LLB/BW

WHERE LLB = number of blocks defined for input image line
BSB = number of blocks covered by rectification line slant
BRL = number of blocks required to rectify the given line
BL = block length in pixels
BW = block width in scan lines
NBRL = numbers of new blocks required per rectified line

Figure 3-1. Disk and Memory Block Organization

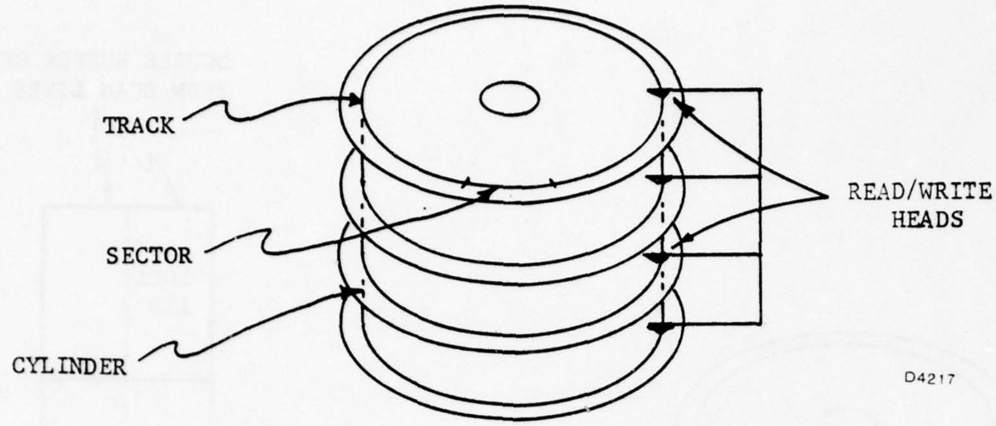
Clearly the block organization will reduce the amount of memory necessary to achieve any rectification or stereo matching situation. In addition and equally important is the reduced number of accesses required to obtain the required data from the disk memory device. This is important because a large percentage of the time necessary to access information on a disk is related to seek and latency times. Typical disk characteristics are shown and described in Figure 3-2. As shown, seek and latency times are quite large for a typical disk unit. Therefore, it is important that the data is organized on the disk in a manner consistent with the way an algorithm accesses it so that seek and latency times can be minimized.

The ideal situation would be to have all the blocks of a column on one disk cylinder. One entire column could then be accessed with one seek. Moreover, if the data accessing occurs in a column by adjacent column fashion, then only incremental seeks are required, and the incremental seeks are faster than both average seeks and full seeks.

Conceptually, the image blocking function will operate as shown in Figure 3-3. There will be two buffers each holding a maximum of 128K pixels (64K words). As one buffer is filled, the other will be written to the disk.

The hardware requirements for the image blocking operation are shown in Figure 3-4. The component definitions are as follows:

SC	=	SCANNER AND ASSOCIATED INTERFACE
CP	=	CONTROL PROCESSOR (CONTAINS SMD DRIVER AND FILE SYSTEM)
DCC	=	DATA CHANNEL CONTROLLER AND MEMORY
SMD	=	STORAGE MODULE DRIVE (DISK)
M1	=	BLOCKING MEMORY (BUFFER ONE)
M2	=	BLOCKING MEMORY (BUFFER TWO)



- A track is the trace of a static read/write head on a surface.
- A sector is the minimum addressable portion of a track.
- A cylinder is defined by the same track on many surfaces.
- There is one read/write head per disk surface.
- Disk capacity set by number of bytes per track and number of disk pack surfaces.
- Transfer rate set by the number of bytes per track and the rotational rate of the disk.

  Example: rotational rate = 3600 RPM, bytes/track = 20,160

  $$60\ \frac{\text{rev}}{\text{sec}} \times \frac{20{,}160\ \text{bytes}}{\text{rev}} \times \frac{1\ \text{M byte}}{10^6\ \text{bytes}} = 1.2.\text{M byte/sec.}$$

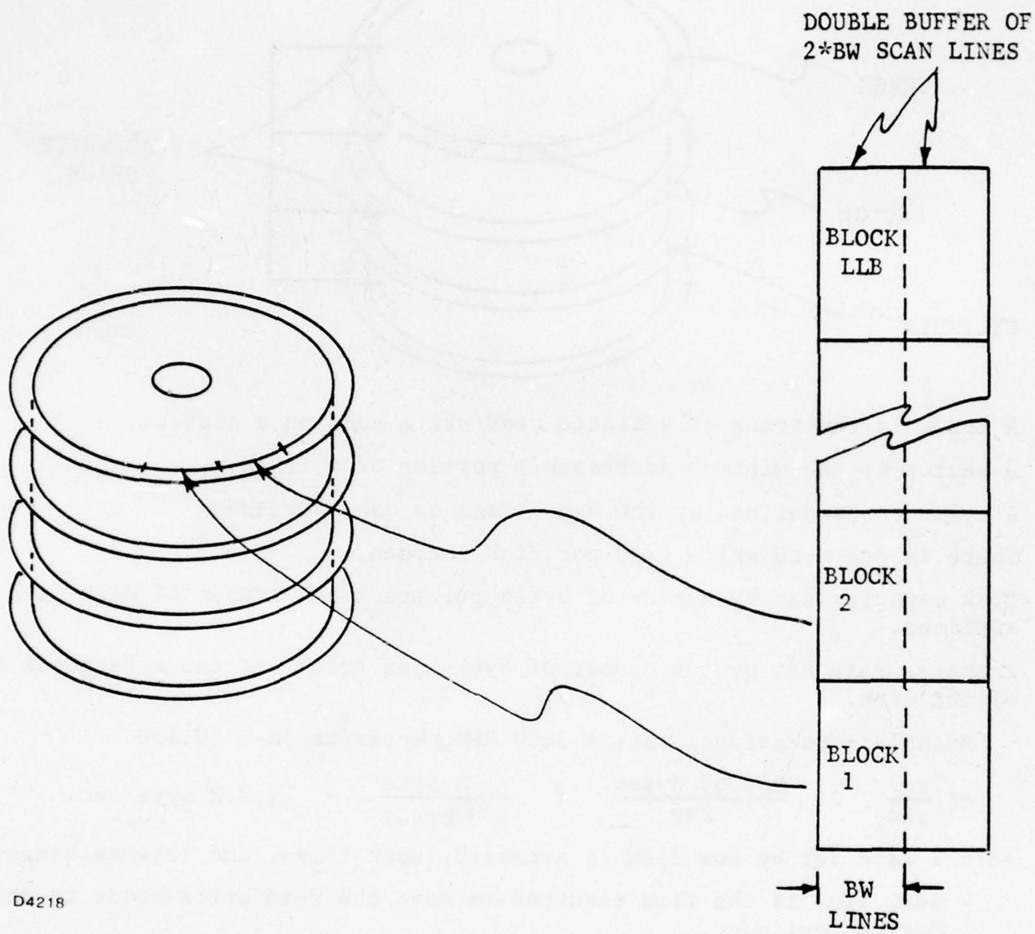
- Actual rate set by how disk is accessed, seek times, and latency times.
  - Seek time is the time required to move the read/write heads to the desired cylinder.
  - Latency time is the time required for the disk to rotate such that the desired sector is under the read/write head.
- Representative times:

Maximum seek time	55	msec
Average seek time	30	msec
Maximum latency time	16.8	msec
Average latency time	8.4	msec

Example:

Maximum rate (no seeks)	1.21 Mb/sec
Rate with incremental seek/cylinder	1.08 Mb/sec
Rate with average seek/cylinder	.89 Mb/sec
Rate with maximum seek/cylinder	.76 Mb/sec

Figure 3-2. Disk Conventions



- Each block recorded on disk sector
- As blocks in buffer one are written to disk buffer two is filled

Figure 3-3. Image Blocking Concept

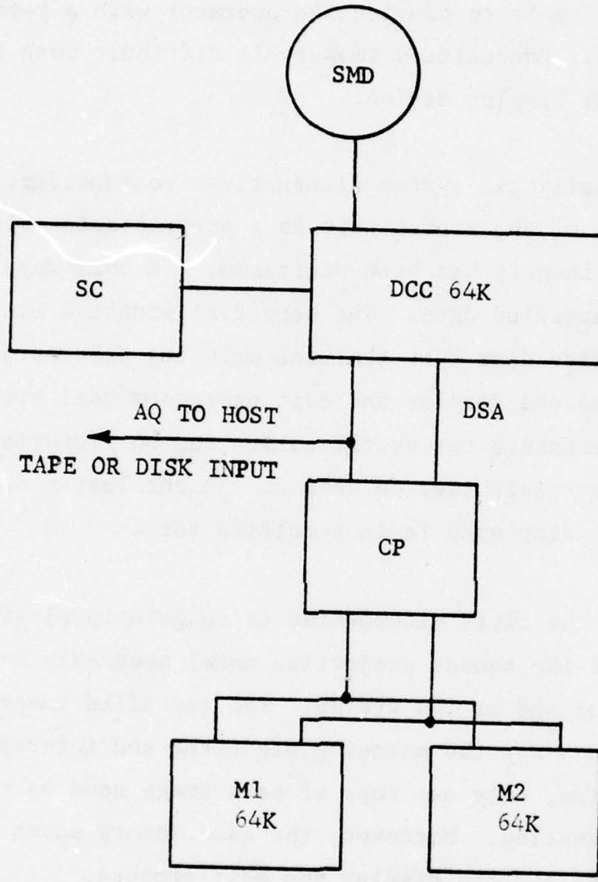


Figure 3-4 Hardware Requirements for Blocking

3.3 Image Rectification

A major question which arose in the initial design of the DMS was whether or not to rectify the panoramic image material before processing. The primary motivation for rectifying is to provide the operator with a y-parallax free display of the imagery. Unrectified imagery is difficult both to view and to edit in stereo on a CRT display device.

There are essentially two system alternatives to consider. The first is to rectify both images of the stereo pair as a preprocessing step that occurs immediately after the imagery has been digitized. In this way, all subsequent processes deal with rectified data. The second alternative is to handle both unrectified and rectified data such that the matching process deals with unrectified imagery and the display and edit processes deal with rectified data. Under this alternative the rectification can be performed in total as a preprocessing step or piecewise, on demand. In the latter case, only the image window currently displayed is in rectified form.

The advantage of the first alternative is computational simplicity. The full complexity of the sensor projective model need only be applied once, at the very front end of the system. The rectified image represents a simpler geometric form for the matching algorithm and intersection routine to process. In addition, only one copy of each image need be retained in the system during processing. Moreover, the same memory space may be shared by the matching process and the display and edit process.

The disadvantage of rectifying the imagery as a preprocessing step is that the imagery must be successively resampled. When the imagery is scanned, it is sampled, and a slight loss in resolution can occur depending on the scanning parameters. This digital imagery then undergoes a rectification transformation which involves a resampling of the already sampled imagery. Finally during matching, Image B of the pair is resampled again in the patch shaping process. The net result is that the original image has been sampled and twice resampled before match point determination has taken place.

The advantage of the second alternative over the first is that one resampling step is eliminated. Unrectified imagery is processed directly by the matching algorithm. The matching algorithm can indeed handle unrectified imagery provided that the panoramic sensor model is included in the prediction and shaping mechanism of the algorithm. Thus, implicit rectification is performed in the analytics of the process rather than as a separate image processing step.

The disadvantage of the second alternative is that a separate rectified copy of the imagery must be kept by the system for display and edit purposes. The storage required for this second image copy can be minimized by employing the rectify-on-demand approach mentioned previously. But the penalties to pay here are slower image display times, and the constraint that when an image area is redisplayed, it must also be rerectified. An additional problem with this alternative is that match points are generated in unrectified space yet match-point editing is performed in rectified space. Therefore, coordinate information must be constantly transformed from one space to the other.

From a systems point of view, the first alternative is the most attractive because of the computational efficiency involved. Thus the tradeoff to consider is computational efficiency vs. match point accuracy. It is unknown at this time the extent to which accuracy is compromised in the first alternative. Therefore, it is recommended that an image processing study be undertaken to determine what the effects of rectification are on the match point accuracy for a variety of resampling criteria.

When digital rectification is to be performed by the DMS, it will be performed with respect to the airbase. In this way, epipolar lines will be parallel to each other and parallel to the digital raster of the imagery. Either the POTS triangulation model or a standard panoramic chimney-type model will be used in the analytics to drive the rectification process.

One additional design consideration that must be kept in mind is that the rectified version of a panoramic image section is always larger in format than the original unrectified section. Thus, there are more pixels to store in the temporary data base and more pixels to process in automatic matching and interactive editing.

3.4 Stereo Matching

During Phase C of this ongoing contract, a benchmark version of the stereo matching algorithm was actually coded and run on an array of Flexible Processors. That version was designed and coded for a maximum of four Flexible Processors. Since that time, enhancements to the matching algorithm have been completed showing improved performance. Using the information gained from the analysis of the benchmark system's performance and allowing for algorithm enhancements, a new parallel array configuration has been designed. That design will be the subject in this section.

Before explaining the components of the matching array, a review of the expected image data flow through DMS is in order. Figure 3-5 shows the relationship the matching module has with the other system modules. After scanning and rectification, the image data will be matched. The image files will be divided into swaths and matching will be completed on a swath-by-swath basis until the specified area is completed. The division into swaths is for two reasons. The amount of memory needed to hold the A and B image match windows in the matching module is reduced and secondly the amount of data which can be usefully displayed for editing is limited.

The matching module components are shown in Figure 3-6. The functions performed by each component are as follows:

VIA
- Serve as link between host and remaining array processors and peripherals
- SMD file system and driver
- Move image data from DCC memory to process memories

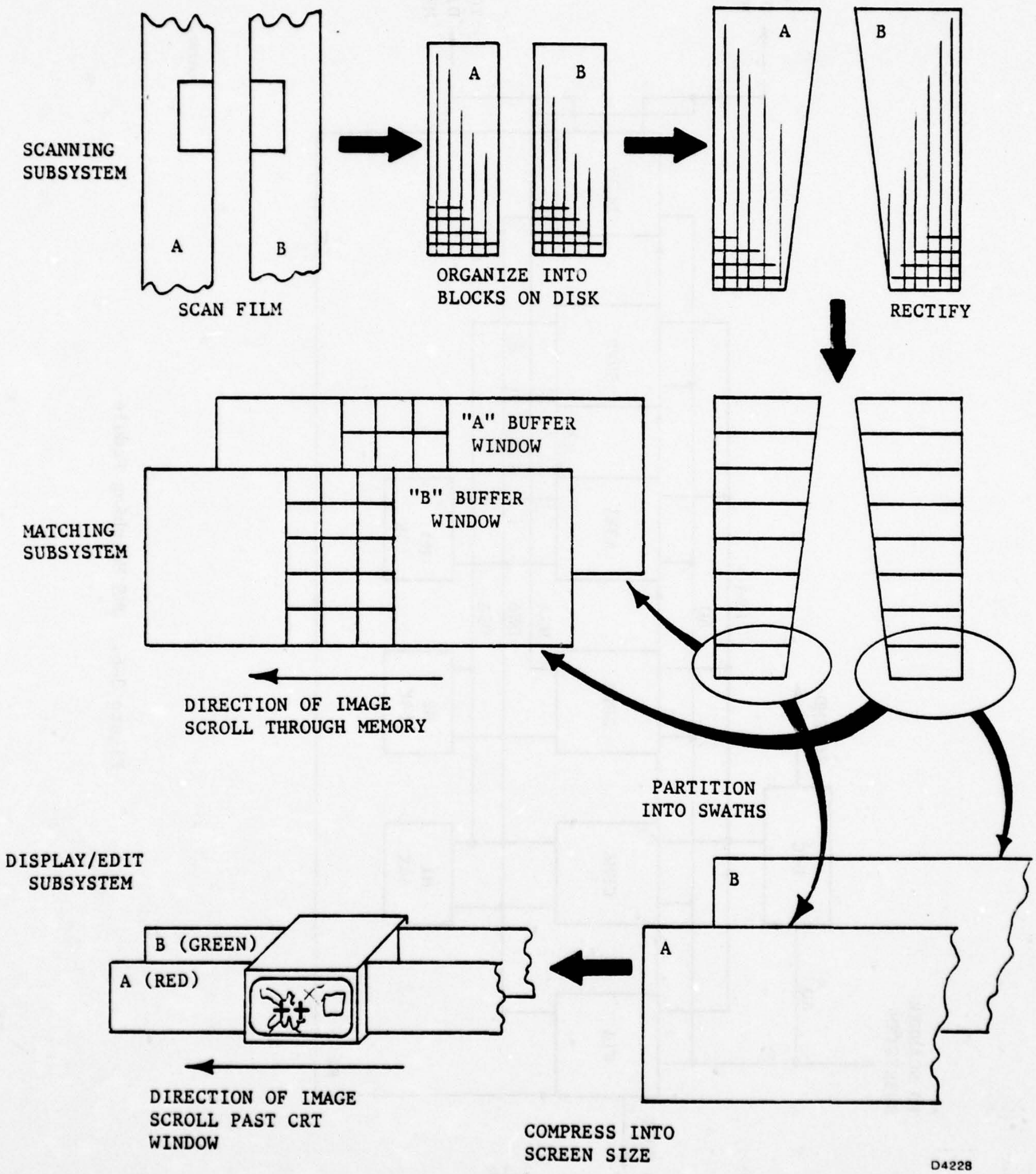


Figure 3-5. Image Data Flow Through DMS

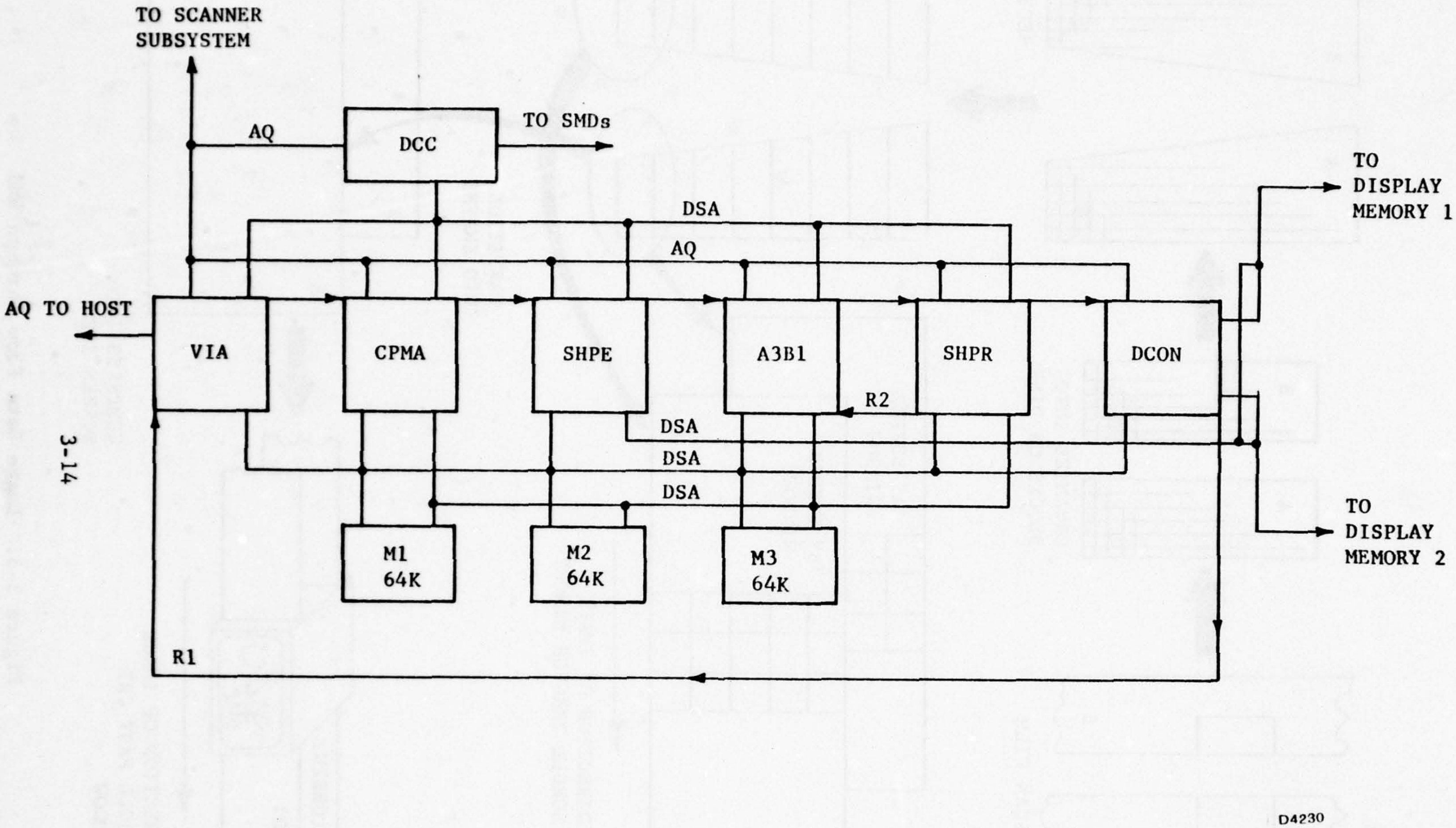


Figure 3-6. DMS Matching Module

CPMA • Serve as control processor for matching

• Initiate matching by sending patch data to shape processor

• Compute cross correlation metric for search sites with summations supplied by other processors

• Refine predicted match point

• Correct match point if warranted (harness)

• Update match file with new point and output point to buffer in DCC memory

SHPE • Use epipolar geometry to define vertical component of match point

• Compute shaping parameters for resampling processors

• Move A image data to display memory

A3B1 • Access A image data and compute patch summations for A image

• Compute inner product summations for correlation sites on row by row basis using A and B image samples

SHPR • Using shape parameters, resample B image data. Compute B image patch statistics for all correlation sites

DCON • Display control program. Interface with two digital scan converter memory subsystems to drive interactive color display monitor

• Move B image data to display memory

• Generate B image patch center lines for visual interpretation of matching performance

M1 • Process memory holding A image matching window

M2 • Process memory holding odd lines of B image matching window

M3 • Process memory holding even lines of B image matching window

R1 • RING number 1

R2 • RING number 2

The time-critical elements in the process are the operations performed by A3B1 and SHPR. The other critical aspect of the design is the time necessary to move the image data from the SMDs to the processing memories. Analysis has shown that the SMDs should be able to support the matching module up to the point where a new match point is generated every 1.5 milliseconds.

The expected performance of the matching module can only be roughly bounded at the present time. To get a better time estimate of the system's performance, some implementation methodology differences must be resolved between the higher level sequential version of the algorithm and the parallel version. The expected performance of the matching module is summarized in Table 3-1. Implementation 1 is essentially the benchmark version with some minor changes. Implementation 2 times reflect a microcode version incorporating all the enhancements existing in the higher level language version of the algorithm.

The other aspect of Table 3-1 which should be explained is the effect of adding processors to the basic matching module. When processors are added to the basic group of six, they are added in pairs as shown in Figure 3-7. As stated before, the time-critical processors in the matching module are the A3B1 and SHPR processors. The system model times are determined exclusively by the time required by those two processors. The system model times are appropriately reduced with the addition of processors doing those functions. The initial software design will allow for expansion of the matching module in this way.

TABLE 3-1. DMS MATCHING MODULE TIME ESTIMATES

NUMBER OF PROCESSORS	MODEL TIME (IMPLEMENTATION 1)	MODEL TIME (IMPLEMENTATION 2)
6	40	200
8	20	100
10	-	67
12	-	50

- TIMES ARE IN MINUTES
- MODEL SIZE IS 3" x 6"
- SCANNING INTERVAL 12.5 MICRONS
- MATCHING DENSITY = 10 x 10 GRID
- TYPICAL CHOICES FOR MATCH PARAMETERS ASSUMED

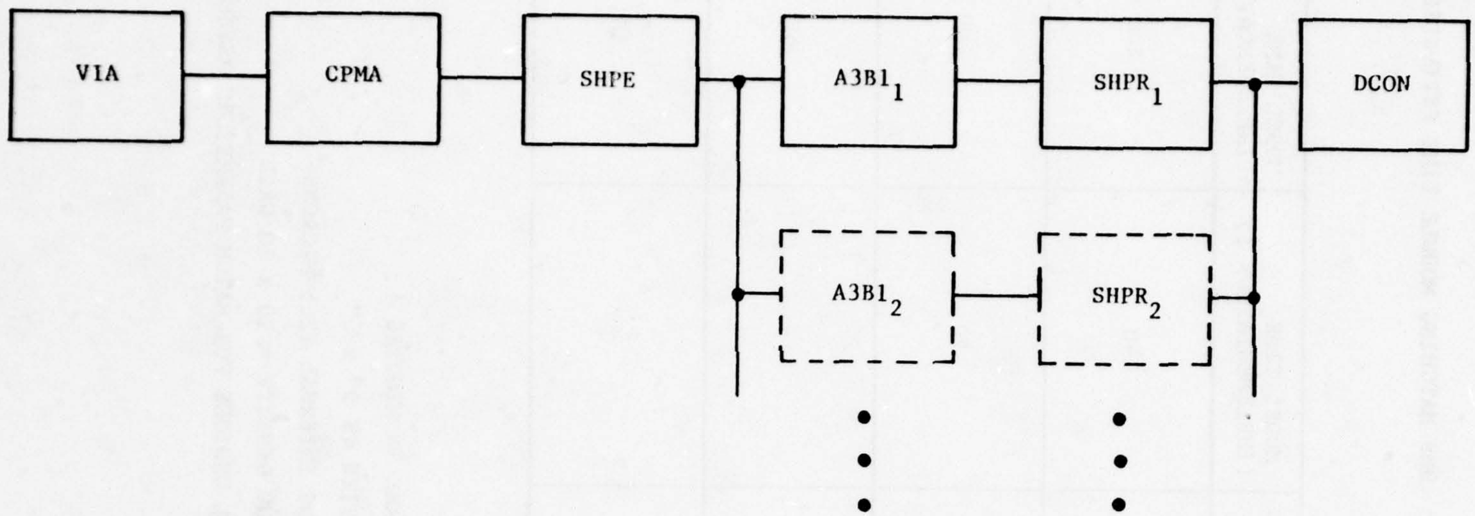


Figure 3-7. Processor Addition to Basic DMS Match Module

3.5 Interactive Editing

It seems that no matter what level of sophistication is attained by any automatic matching algorithm, there will remain regions that cannot be matched accurately enough. If the matching algorithm is a good one, and if enough time can be allotted for tuning of the algorithm's parameters, then the frequency and effect of the hard-to-match regions can be minimized.

Admitting the fallibility of any automatic matching algorithm for some regions immediately implies the need for operator assistance. At what point, then, is it best to provide the assistance and what is the best man-machine interface to provide.

The digital mapping system will allow for edit processing during the match point file generation phase, or on a review basis after the match point file has been completed but prior to any further processing.

3.5.1 Edit Option During Matching

To understand the edit process as it is conceived in this system it is appropriate to review outputs of the matching system and how they will be perceived by an operator.

For each point matched, a 5-tuple is stored consisting of Image A coordinates (x,y) and conjugate Image B coordinates (u,v) along with a n-bit reliability factor. This 5-tuple will be perceived by an operator as follows: A and B image data will be scrolling from left to right across a stereo color monitor as matching proceeds. Superimposed on the image will be an overlay of the lines connecting the A image patch center coordinates (x,y) (i.e. straight lines since in a column of points all (x,y) coordinate values will be the same) and the lines connecting the B image matching patch center coordinates (u,v). Figure 3-8 represents the connected match point overlays. Both overlays will be superimposed on Image A (red) and Image B (green) using a single color monitor such that an operator will perceive in stereo a profile

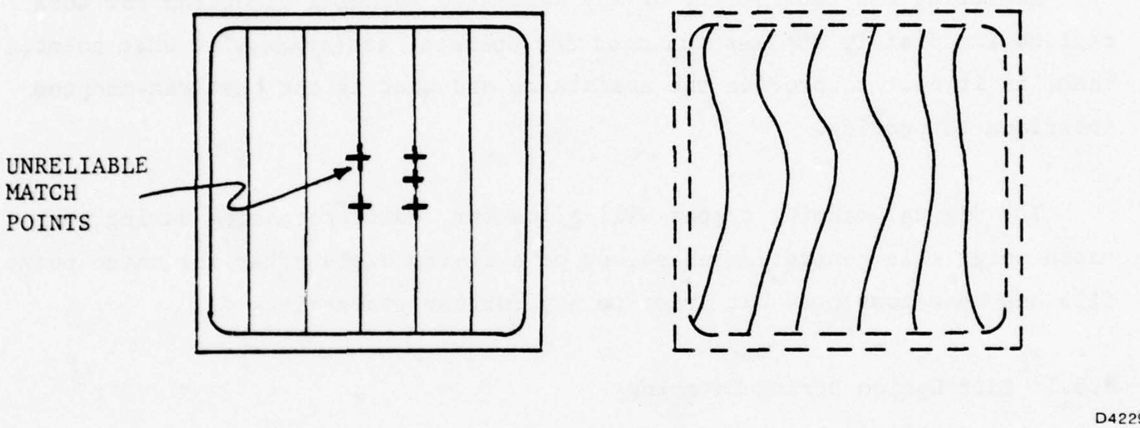


Figure 3-8. Match Point Monitoring on Color Display

following the terrain covered by the column of match points. In order to depict the reliability of each match point, a symbol can be added to the line of one of the images at each point where a match point is unreliable.

When it becomes apparent that the matcher needs some assistance, the operator has two options. First, he can halt the matcher at the end of any column of points. Matching parameters can be changed and the matcher restarted at a point preceding the trouble spot. If this is done, the matcher will reinitialize itself using the existing match point file. The second option is to halt the process and note the region where post-match editing will be needed. This is done by taking a snapshot dump of the display and noting the region of difficulty on the hardcopy output. To facilitate location of these regions, tick marks will be superimposed on the images along the bottom of the screen listing x and y coordinates at appropriate intervals.

3.5.2 Post-Match Edit Options

In a post-match edit mode, the operator will have four basic options depending upon the type and size of the area over which corrective action is required. For this discussion, we will refer to the four options as profiling edit mode, area edit mode, point edit mode, and line edit mode. The four options are conceptually depicted in Figure 3-9.

3.5.2.1 Scrolling Facility

To assist in post-match editing, an operator will be given the option of requesting a scroll function. By specifying a starting location (Image A coordinates), this option will once again present the information observed during matching on a color monitor. The operator will be able to specify the resolution or effective match point density he wishes to inspect. Image A (red) along with Image B (green) and superimposed, connected match point files will be presented. The operator will be able to stop at any point or group of points he wishes to correct and request that the appropriate edit mode be enabled.

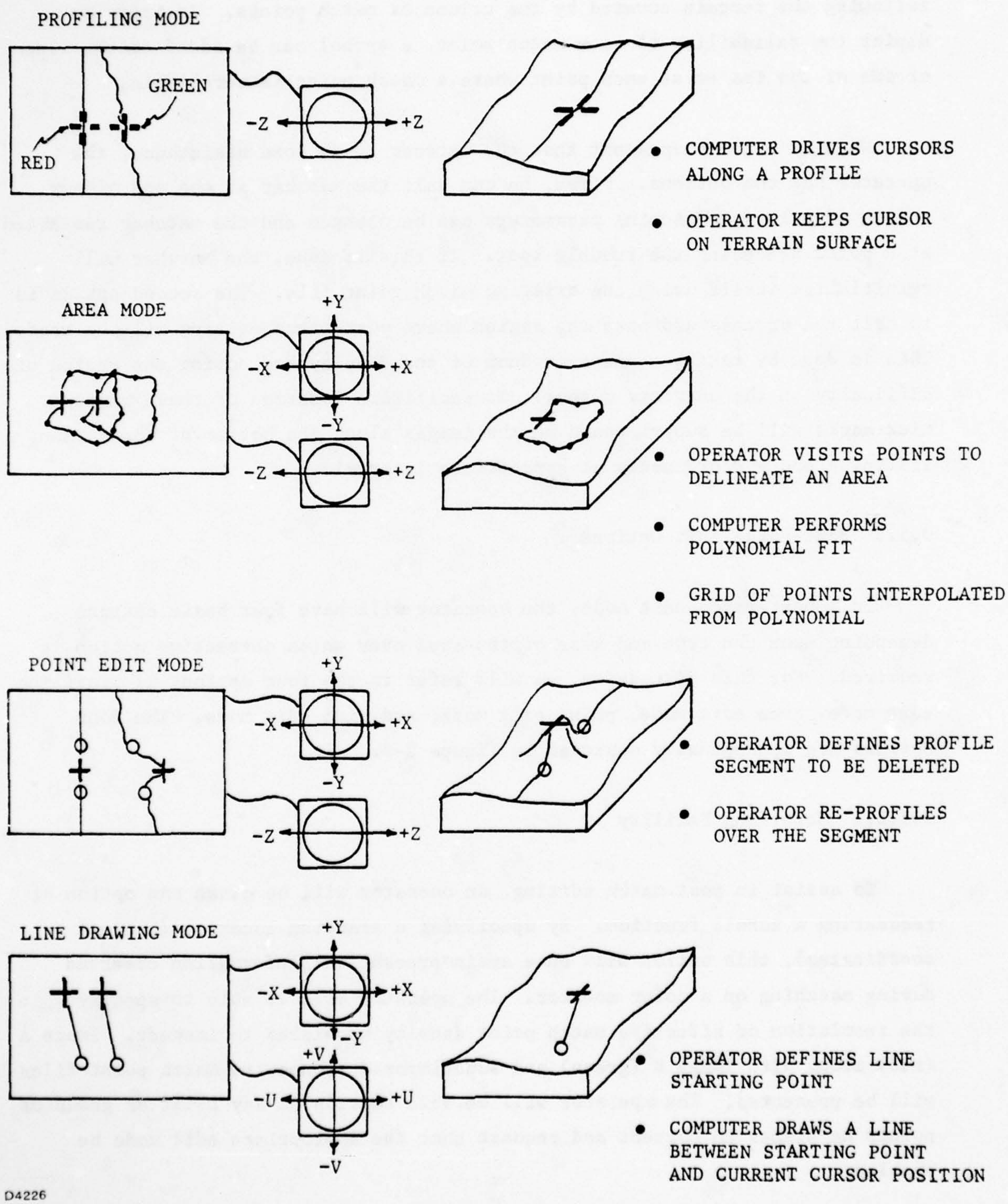


Figure 3-9. Post Match Edit Modes

3.5.2.2 Profiling Edit Mode

The profiling edit mode can be requested when an operator wishes to define an entire column of match points. A pair of cursors, one in red and one in green, will be superimposed on the A and B images respectively. The cursors will have their movement in the y dimension synchronized. As the A Image cursor steps along, the operator will be responsible for keeping the B Image cursor positioned correctly on the virtual terrain surface. As points on the match grid defined on Image A are crossed, the B Image cursor location will be read. The match point 5-tuple (x,y,u,v,r) will be stored as a totally reliable point.

3.5.2.3 Area Edit Mode

The area or region edit mode will be used on larger areas where it would be impossible or impractical to use one of the other edit modes. This may be a large featureless area that simply cannot be matched. In this mode, the operator with the aid of cursors will choose a number of match points around or within the region. These points will subsequently be used to fit a polynomial which will define the transformation of Image A to Image B coordinates. All points lying on the match grid on Image A and within the operator defined region will have new match points generated.

3.5.2.4 Point/Segment Edit Mode

The point edit mode is essentially the same as the profiling edit mode except that it is done over smaller bounds. An operator using the image cursors will be able to redefine single match points or match points within a short segment with this edit mode.

The system will request the operator to define bounds along a single profile over which he wishes to make corrections. This is done by positioning the B Image cursor at an upper and lower point on the match point line overlay. The existing overlay will be erased between these two points. The operator

will then be requested to redraw the line overlay between the bounds. As he is redrawing the line overlay using trackball and cursor, each time the A Image cursor passes a point on the match grid, the B Image cursor position will be read, thereby defining a new match point.

3.5.2.5 Line Edit Mode

The line edit mode is similar to the point or segment edit mode. Once again the operator will be directed to specify on upper and lower bound over which to effect correction. The difference will be, however, that the operator will not have to redraw the match point line overlay. Instead, the system will connect the two bounds with a straight line segment. All match points within the correction bounds will be redefined as lying on the straight line segment.

3.6 Postprocessing Functions

Most of the computational modules of the DMS are directed toward the automatic production and operator-assisted editing of match point data. The primary output of the DMS is a grid of terrain data suitable for inclusion in an on-line data base or for dumping onto magnetic tape. Therefore, two computational steps are required to convert the match point data to the final terrain data grid. These two steps are Analytical Intersection and Grid Normalization. Because of the hardware configuration for the basic DMS functional modules, analytical intersection is considered part of the Stereo Compilation function and grid normalization as part of the Data Preparation function.

3.6.1 Analytical Intersection

This is a straightforward photogrammetric computation that requires no parameter tuning or operator assistance. Match points expressed in pixels are transformed to photo coordinates by the interior orientation transformations and then intersected either in a model or object space coordinate system using

the parameters derived in the triangulation of the imagery. The intersection involves the solution of four collinearity equations in three unknowns for each point. The less rigorous but more computationally efficient method of simple projection from two stations can also be used if the error budget will permit it. If an absolute orientation transformation exists to transform terrain data to a ground truth system, then it is advantageous to include this transformation at the front end of the intersection computation such that the intersection occurs in the final desired coordinate system. This eliminates the computational expense of transforming already-intersected points to yet another coordinate system.

3.6.2 Grid Normalization

This is the last computation to be performed before terrain data can emerge from the system as a final product. The matching process is driven by a regularly spaced grid of points on Image A. When these points are intersected, the result is an irregularly spaced grid of terrain data. Thus, the purpose of grid normalization is to generate a new regular grid with the required spacing from this irregular grid. The irregular grid is much more dense than is required as output, due to the nature of the matching process. Therefore, linear interpolation of the final grid seems to be a sufficiently accurate approach. The reliability factor of each normalized grid point is likewise interpolated from the neighboring reliability factors of the irregular grid. The normalized grid can be created either on the system disks, for temporary storage, or output directly to magnetic tape.

3.7 Display Generation

If the operator is to be efficient, functioning as another processor in the DMS system, then he must have at his disposal adequate interactive tools. The eventual throughput rate of the DMS will be determined to a great extent by the amount of time spent in the interactive modes. Therefore, the better the interactive display tools, the more efficient the operator, and the higher the total throughput rate of the system.

Digital image processing offers a wide range of display capability. The actual image display device, even though it is small in size (512 lines by 640 pixels) compared to the entire image format, functions as a window into the system memory. Thus, any pixel arrangement that can be created in memory is displayable. This includes image gray-scale information as well as graphic and annotative information, and operator-generated overlays. In addition, stereo capability exists by virtue of the fact that the system includes two scan converters. Thus, two images may be displayed independently on the same CRT, colored differently for an anaglyphic stereo presentation.

During production stereo compilation, the primary purpose of the display station is to provide the operator with diagnostic information that indicates how well the automatic process is performing on the imagery. However, in the editing phase, the role of the display software is to assist the operator in making fast and accurate corrections to ill-compiled data. It is unclear at this time what display formats are optimum for the various interactive modes. This may become clear only after real operators have experienced the system in a real operating environment. Therefore, the intent in the DMS design is to provide several different display formats that can be experimented with and evaluated in terms of their production usefulness. The following sections describe the display formats that are candidates for inclusion in the system.

3.7.1 Original Image Display

The most fundamental display of the DMS system involves the organized presentation of the original or rectified imagery that the system operates on. This display provides the base upon which all subsequent graphic overlays can be superimposed. The automatic matching of the stereo imagery proceeds in swaths over the data. The width of the swath can be adjusted to correspond to the number of pixels that can be displayed in compressed form in one dimension of the CRT. The two scan converters are utilized such that each eight bit image of the stereo pair is handled by one scan converter with its associated display memory. One scan converter projects an image in red, the other in green for anaglyphic stereo.

As the matching process proceeds along the swath, the displayed imagery is made to scroll past the display from right to left. This is accomplished by successfully altering the scan converter index tables. The scrolling rate can be adjusted by the operator; however, it is advantageous to make the scrolling rate consistent with the image matching rate, particularly if operator corrections or restarts are to be made during the matching process.

For scanning and rectification process review, the images may be displayed separately and the scrolling can occur in either orthogonal direction. In addition, any desired window of the imagery may be displayed in full resolution on a demand basis. However, this display mode is a bit slower than the in-process scrolling display because the desired image window must be brought from disk into the display memory before it is displayed. The reason for compressing the data by averaging in the in-process display is to provide larger area coverage for review purposes.

3.7.2 Match Point Plot

The direct product of the automatic matching process is a file of match points. The relationships between match points can be presented in a graphic display by digitally connecting adjacent match points in one image direction to produce match point profiles. These profiles then can be superimposed on their respective image. Since the matching process is driven by a regular grid of points on the image denoted on Image A, the profiles on this image appear as truly straight lines. However, on Image B the match point profiles are not straight lines; the higher order bends in the profiles are in direct proportion to the terrain relief. Thus, the profiles visually represent the parallax function that exists between Image A and Image B. Examples of these match point plots are shown in Section 6.0 of the Fourth Interim Technical Report.

When the match point profiles described above are superimposed on their respective image and displayed in stereo, the conjugate profiles fuse stereoscopically and appear to lie on the surface of the terrain in three dimensional

space, where matching has been accurate. Inaccurate match points are portrayed by profile segments that appear to lie above or below the terrain surface. This display then is visually equivalent to hundreds of floating marks in a conventional instrument connected in profile fashion.

The operator can directly edit match point data using this type of display. In the edit mode, the system generates a stereo cursor that the operator controls with trackballs. This is in fact a digital floating mark. Thus, the operator can, by use of the cursor, delete erroneous data and manually recompile correct data. The operator performs the editing in terms of three dimensional stereo space, where he perceives the cursor, but the system actually makes the correction to match points in terms of two dimensional image space. In this way, the display and edit functions are match point oriented, making them compatible with the automatic matching process. In addition, the need for constantly intersecting and reprojecting points between image space and object space is eliminated.

3.7.3 Reliability Plot

A number of display formats offered in the DMS for operator viewing are intended as diagnostic displays. That is, the operator can, at a glance, determine whether and where the automatic processes are faltering. One such display is the reliability plot. An integral part of the automatic matching algorithm is the internal reliability monitoring of the data that is produced. As is detailed in the Fourth Interim Technical Report, a reliability factor is computed for each match point generated. The collection of these quality measures then can be gray-scale encoded to indicate the degree of unreliability and composed into an overlay which can be displayed along with Image A. An example of this reliability plot is included in Section 6.0 of the Fourth Interim Technical Report.

Unreliable match points tend to be clustered in groups. Therefore, these groups appear as blob-like features on the reliability plot. The problem then is that these features obliterate the image background when the plot is

overlayed on Image A. It is precisely these image areas that the operator may wish to analyze. The solution to the problem seems to be to tone down the reliability presentation such that unreliable points are indicated with small symbols such as circles or tick marks that do not obliterate the image background. It may also be more advantageous to include these unreliability symbols in the match point plot discussed previously rather than retain a separate reliability plot.

3.7.4 Tonal Difference Image

Another diagnostic display format is the tonal difference image. This image is produced by spatially transforming Image B of the stereo pair such that it is in registration with Image A. The transformation is based on the conjugate match points. Then the transformed Image B is digitally subtracted, pixel by pixel, from Image A. When the images of the stereo pair are similar radiometrically and when accurate matching has occurred, then this difference image is theoretically zero. For display purposes, these zero pixels can be assigned a value corresponding to middle gray. Then positive differences will appear darker than middle gray and negative differences will appear lighter. If this tonal difference image is also enhanced or color coded as it is displayed, then the areas of poor matching become readily apparent to the operator. An example of such a tonal difference image appears in the First Interim Technical Report.

This differencing technique to detect poor matching can also be employed in a stereo display mode. Here, Image A and the transformed Image B are displayed anaglyphically as if they were a stereo pair. The areas of accurate matching will appear as part of a flat reference plane in the three dimensional model space. Poorly matched areas appear as elevations or depressions with respect to this reference plane.

3.7.5 Elevation Contour Image

An elevation contour image can be generated digitally only after the match points have been intersected to model or object space to produce X, Y, Z terrain data. The terrain data is then fit with local surface modeling functions, typically polynomials, and the contour line positions are interpolated from these polynomials.

An example of a digital contour image appears in the Fourth Interim Technical Report. The particular method used for that image involves the construction of bicubic surface polynomials that fit the data exactly in a local region of validity. Thus, no data smoothing has taken place and the contour lines are representative of the original match point data. Where there is a poorly matched point in relation to its neighbors, there is a corresponding irregularity in the contour lines in the vicinity of that match point.

The important factor to consider is that the contour image is an interpolated product. It is not a direct portrayal of the match point data or the actual terrain points. Even though the contour image can be used for diagnostic purposes, it is dubious whether it can be used effectively for editing purposes. The contour lines can indeed be redrafted to correct apparent errors by the operator using the cursor in a delete and insert mode. However, to alter the actual terrain data grid or the match point data on the basis of corrected contours presents somewhat of a computational problem. The system must determine from the redrafted line how the terrain points in the vicinity of the line are affected. This involves a reverse interpolation by iteration, a computationally complex and time consuming process.

The contour image can be produced in unrectified image space, as in the example cited above, or in model or object space. The advantage of producing contours in image space is that the contour lines may then be superimposed on Image A to provide a background feature reference. To obtain such a presentation in model or object space requires that Image A be differentially rectified

to the orthographic terrain grid. This is all possible digitally, but the computational expense makes it rather impractical, considering the fact that orthophotomaps are not end products of the DMS system.

3.7.6 Three Dimensional Plot

The intersected terrain data can also be displayed graphically in three dimensional form from any viewing aspect. Actually, what is displayed is a two dimensional terrain grid, but the viewer perceives it in three dimensional perspective. An example of this type of plot is contained in the First Interim Technical Report.

Like the contour image, this display form is a derived product, and as such its usefulness as an editing tool is limited. However, it does give the operator a feel for what the terrain actually looks like from various aspects. In addition, spikes and other irregularities due to poor matching become readily apparent in this type of display.

3.7.7 Profile Plot

This plot is similar to the three dimensional plot described above. Here, however, only the X-Z or Y-Z dimensions of the intersected terrain data are plotted; no spatial projections are made. Terrain profiles are plotted adjacent to one another on the display by merely connecting the points along a profile by straight line segments. Deviations along the plotted profile from a straight line are directly proportioned to the elevations above the datum along the real terrain profile. The net visual effect is that the collection of plotted profiles resembles the terrain shape. This type of plot has been used effectively in the past by DMA to detect spikes in the terrain data and as a tool for smoothing the data on a profile basis.

4.0 SYSTEM DIAGRAMS AND HARDWARE

The functional block diagram of the DMS introduced in Section 2.0 is translated into a hardware block diagram that could be used to implement the DMS capability thus far described. Two levels of system capability are defined. The first, discussed in Section 4.1, is a full capability version intended for production work. Improved production throughput can be achieved by replicating parts of the hardware to provide multiple user capability. The second, discussed in Section 4.2, is intended as a validation system. All capability of the full configuration can be demonstrated and tested but not simultaneously. When the image scanner is used with the validation system, for example, no other Data Preparation or Stereo Compilation functions can be performed at the same time.

4.1 Full Capability Configuration

The DMS functional capability illustrated in the block diagram, Figure 2-1, is implemented with the hardware configuration illustrated in Figure 4-1.

4.1.1 Hardware Elements

Hardware elements which appear in the diagram are MOS Memory Modules (Mem), Flexible Processors (FP), and Data Channel Controllers (DCC). Also appearing in the diagram are blocks representing a Host Computer, User Terminals for data entry, the Image Scanner, a Stereo Display and a Color Monitor, both equipped with Interactive Controls, and a Dual Port SMD Bank for data storage. An SMD (Storage Module Drive) is a large capacity, high speed disk unit.

The Flexible Processor, the MOS Memory Modules, and the Data Channel Controller are from a family of Digital Image Systems Division products. They are specifically designed for use in reconfigurable arrays for real time and rapid processing of image data. The Flexible Processor is a microprogrammable parallel processor with multiple port access to large banks of fast memory, (100 n sec), and high speed data transfer. The high speed data transfer

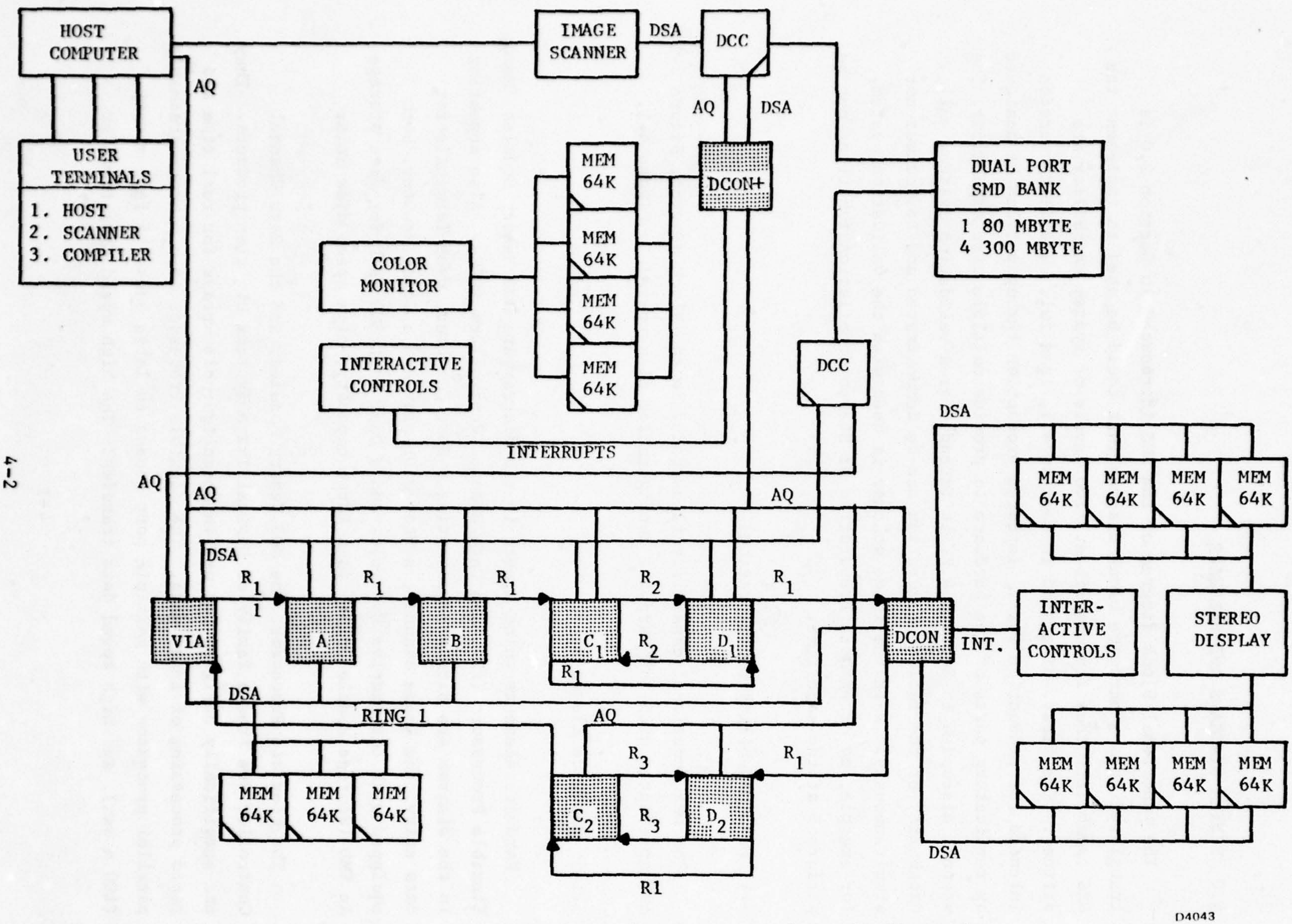


Figure 4-1 Digital Mapping System Hardware Configuration

channel operates at 8 MHz, the cycle time of the FP CPU. Data packets of two 16-bit words each are transferred from one FP to another at this rate; the high speed I/O channel is called a ring channel. A data packet is entered on the ring with a unit destination code and is passed from one FP to the next until the destination unit recognizes its own code, and accepts the data packet.

The FPs in the block diagram (Figure 4-1) are connected to each other with ring channels. The algorithm performed on the data is strung over several fast computing elements in pipeline fashion, each one performing a part of the computation. The FPs have access to the memory units via one or more DSA channels (Direct Storage Access).

The MOS Memory Modules are designed to support the FP computations by providing temporary fast access storage for a region of image data required by the algorithm process.

One FP, labeled VIA, has the task of keeping the Memory Modules filled with currently useful data. The Data Channel Controller, the third of the DISD products, is a memory module equipped with an active station port to process FP requests for data which resides on an SMD disk unit. The DCC includes a line buffer memory (LBM) which is loaded with the disk data requested by an FP. When the FP reads the data in LBM, the DCC is free to reuse that portion of LBM for other data transfer requests.

The Host Computer in the block diagram, Figure 4-1, is used as a control device to initiate and monitor the operations performed by the array of Flexible Processors and associated equipment. FORTRAN programs are written to request, accept, and process user data entries. The User Terminal also displays process status indicators and diagnostic messages. The Host Computer, in response to user requests, loads appropriate microcode into the Flexible Processors, and controls the performance of the Image Scanner.

4.1.2 Data Preparation Hardware

The Image Scanner operates in off-line mode with respect to the Stereo Compilation function. A single scanner is used to operate on a single image. The Color Monitor, with a cursor among the Interactive Devices, is used to display the digitized image and identify pass points, pug points, or fiducial marks. The digitized image is then processed to find interior orientation parameters and blocked for efficient access and stored on the disk. A stereo pair of images, related by previously determined exterior orientation parameters, is processed in this way.

One of the four 300 Mbyte SMD disk units from the Dual Port SMD Bank is used to store the unrectified image pair. Unblocked scanner data is first entered into the SMD Bank. The FP which performs Data Preparation functions is labeled FP DCON+ in Figure 4-1. The unblocked scanner data is read from the disk unit, blocked by DCON+, and rewritten into the SMD Bank. Data Blocking is performed while Image Scanning is in process and the two operations are completed almost simultaneously. Unrectified data for each image of the stereo pair is taken from the SMD Bank; rectified image data is returned to the bank. The rectified image data generated for the particular relative orientation of the stereo pair is organized to permit simple retrieval and display of conjugate regions of the stereo pair.

The hardware elements thus far mentioned, aside from the Host Computer and User Terminals, are provided to support the Data Preparation functions of Figure 2-1. The four banks of MOS Memory Modules, which lie between the Color Monitor and DCON+, are provided to serve as process memory for the Data Blocking, Image Rectification, and Grid Normalization functions. All of these functions, described in Section 2, are performed by DCON+ and make use of the large memory units during the process. The Color Monitor is used as a utility device to inspect the digitized data and to locate points in the image display which have known image coordinate positions in the original material. The bank of four memories also supports these utility

display functions by serving as a scan converter. Each memory has multiple port capability. In this configuration each bank has a DSA port to give memory access to DCON+ and a video port to give memory access to the Color Monitor. The four video ports operate together to form a scan converter which provides video data for display. This approach to image display permits an FP with access to the scan converter memory to perform image processing operations while the data is being displayed. From another point of view, the image display operates without a dedicated image storage medium for maintaining the display. The display operates by looking in on the process memory as operations are performed.

4.1.3 Stereo Compilation Hardware

The Dual Port SMD Bank is the interface between Data Preparation hardware and Stereo Compilation hardware. A rectified stereo pair written on one of the four large disk units is the primary input for stereo compilation. The corresponding unrectified image pair is no longer needed and the disk space which contains it can be released for other use when rectification is complete.

Stereo Compilation hardware has access to the rectified stereo pair by way of the FP, VIA, which performs data file management, and the second of the DCC pair which use the dual port capability of the SMD Bank. VIA maintains the necessary conjugate regions of the stereo pair in the process memory bank at the lower left of Figure 4-1.

The Stereo Compilation algorithm is distributed over the FPs A, B, C, and D. The C and D units are responsible for performing oeprations which pace the entire process. The operations are therefore shared equally by C_1 and C_2 and by D_1 and D_2. The throughput obtained in this way is twice that of C_1 and D_1 alone or a 100% improvement. A further improvement of 50% could be obtained by adding a third pair of functionally similar FPs, C_3 and D_3. The addition of each such pair results in a further improvement, albeit less cost effective.

The FP, DCON, also has access to the process memory used to store the conjugate image data used by the Stereo Compilation Algorithm. Upon completion of the Image Matching function, DCON constructs smaller scale compressed images of the match point region for stereo display. A bit is set in each image at the match point position to create the appearance of a floating mark in the stereo display. DCON, which stands for Display Control, maintains the scrolling Stereo Display by continuously adding new imagery with appended match points to the two memory banks of the scan converters. During automatic image matching the operator views a stereo display with just completed match points, superimposed on the model, entering the display on one side while the oldest ones leave the other.

In fully automatic mode the Interactive Controls affect only the display and do not influence the match point selection. Display functions include zooming and scrolling around within the scrolling image to examine particular areas at a larger scale.

The automatic mode can be interrupted when the match point selections of the automatic process become unsatisfactory. In such cases the Match Point Editing capabilities make use of the Interactive Controls to provide the operator with a variety of tools to correct results, perform manual compilation, perform interpolation, and to redirect, restart, and control the resumption of automatic Image Matching.

The final operation among the Stereo Compilation functions is Match Point Intersection. The match points are intersected in the model space produced by the exterior orientation parameters entered in the system. The 3-dimensional model coordinates are transferred to the Dual Port SMD Bank where they are stored on a disk unit for final processing by the Data Preparation functions.

Image Matching is the Stereo Compilation operation which requires the greatest computer power. The FPs and memory provided for Image Matching therefore suffice also for the other operations.

4.1.4 Host/Control Module Hardware

Hardware elements of the Host/Control Module, corresponding to those of Figure 2-1 (DMS Control and DMS Data Base), include the Host Computer, the User Terminals, and the Dual Port SMD Bank. The functions performed by these hardware elements are evident. The choice of one 80 megabyte and four 300 megabyte disks for the SMD Bank, which contains the DMS Data Base, permits the accumulation of a small backlog of rectified stereo pairs by the Data Preparation hardware while the Stereo Compilation hardware is in operation.

4.2 Validation System Configuration

At some point during the development of DMS capability, it may be desirable to provide a hardware configuration which can be used to test, evaluate, and refine the software and the operational flow plan before commitment to a full configuration. The intent would be to ensure satisfactory performance of the final configuration by demonstration and evaluation of a less expensive system which can be available at an earlier time. To achieve full benefit from the validation system, the hardware elements and much of the software should be used in the final configuration.

A hardware configuration responsive to this requirement is illustrated in Figure 4-2. The functional capabilities of the validation configuration are the same as for the full system. The difference lies in the ability to perform Data Preparation and Stereo Compilation functions simultaneously. By sacrificing this feature of the full production configuration, the functional performance projected for the full configuration can be tested, evaluated, and revised with less hardware commitment. Because of the basic similarity of the two configurations, the software provided to implement the validation system will transfer to the full system very well. Some changes will be

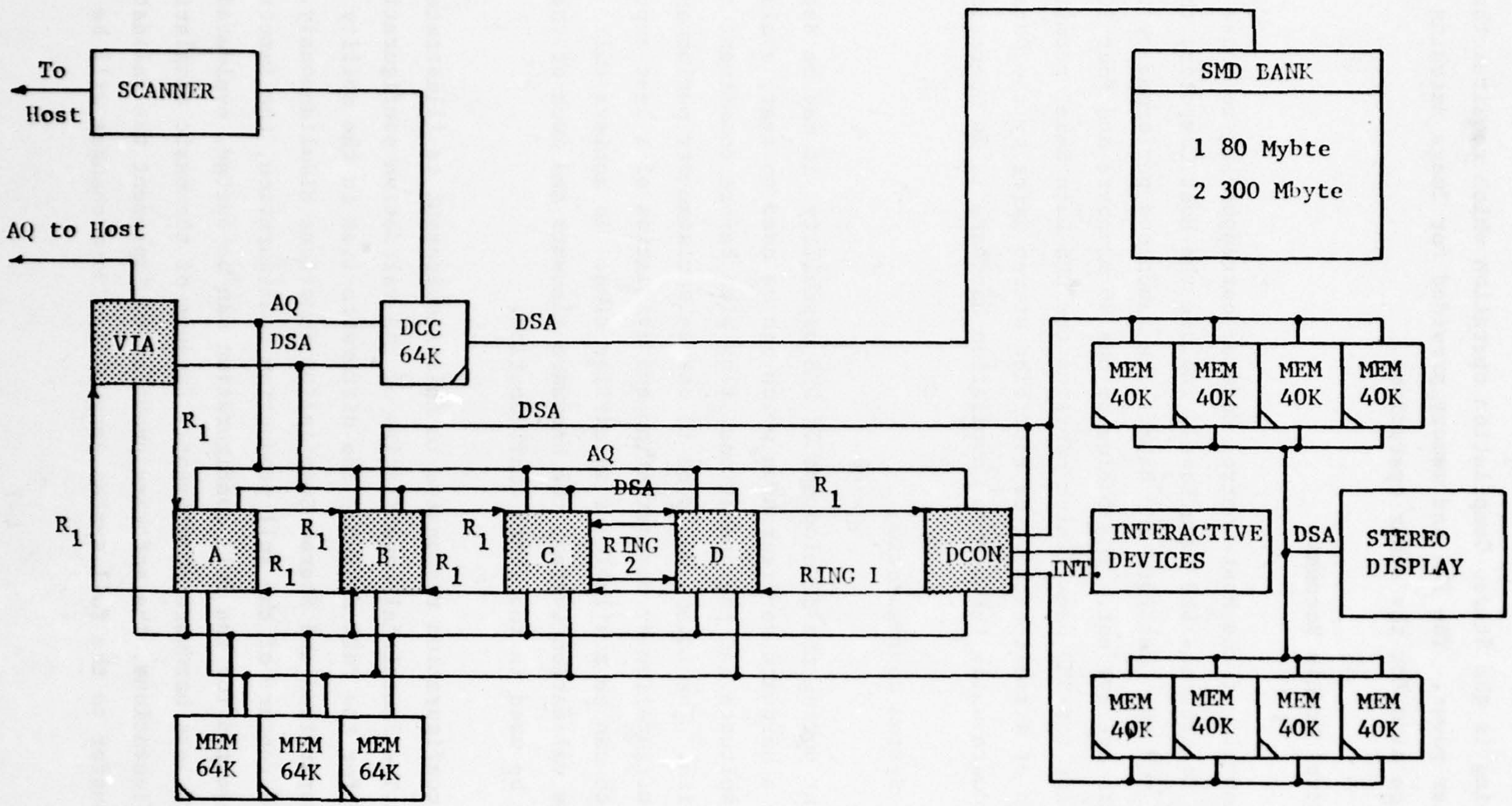


Figure 4-2 Digital Mapping System Hardware Validation Configuration

D4305

necessary for two reasons: the larger system will have more hardware, imposing a requirement for equipment and destination codes unknown to the original software; experience with the validation system also may suggest changes which have an impact on the software.

Differences between the two hardware configurations, Figures 4-1 and 4-2, are in the number of Flexible Processors, Memory Modules, Data Channel Controllers, disk units provided, and interactive displays provided. The Host Computer and a single User Terminal, although not shown in Figure 4-2, are required. Two User Terminals are provided in the full system while one is sufficient for the validation system.

Table 4-1 is a detailed comparison of the hardware elements of the two systems. The software elements of the two systems are the same.

TABLE 4-1. HARDWARE ELEMENTS OF THE DMS AND THE DMS VALIDATION CONFIGURATION

HARDWARE ELEMENT	FULL	VALIDATION	REDUCTION
Host Computer	1	1	0
User Terminals	2	1	1
Flexible Processors	9	6	3
Memory Modules	15	11	4
Data Channel Controllers	2	1	1
Interactive Displays	2	1	1

5.0 SYSTEM PERFORMANCE

The primary objective of the DMS design set forth in this report is to enhance the total model throughput time. By "total" is meant the time it takes for a group of cooperating processes to reduce raw data on film to acceptably accurate digital terrain data.

In the current DMA production process, there are many independent processes that are applied to the data in segmented fashion to produce the final digital product. The processes span the range from automatic compilation, to semi-automatic fill-in, to manual drafting. Much time is spent in the manual communication among and the staging of these processes. In addition, the particular model of interest must be set up repeatedly to obtain a complete data set. Also, since the manual drafting for error correction is applied to the contour manuscript itself, the data corrections are unrecoverable in digital form.

It is felt that throughput enhancement on a model-by-model basis can occur by integrating the various automatic and manual steps within the same digital system. In this way, the product that emerges from the system as a digital terrain model is as complete and accurate as possible, given the limitations of automatic compilation and the extent of human interaction capability.

5.1 Levels of Parallelism

In any kind of interactive system, it is inevitable that the machine will have to wait for the man at several places in the processing stream. However, it is not optimum in terms of system performance for the man to repeatedly wait for the machine. Therefore, a major factor in the DMS design is that when the machine is computing on its own, it is as fast as possible.

This speed is attained in the DMS design because of the inherent parallelism of the system. The three basic levels of parallelism involved are discussed in detail in the Third Interim Technical Report. Briefly, these levels are: the instruction-logic level where many operations can be performed simultaneously in a single instruction cycle; the processor level where distributive processing is implemented by partitioning the algorithms among the processors such that each processor performs its given task simultaneously with the other processors and tasks; and the systems level where the algorithms are duplicated in several processing channels such that the channels operate simultaneously on different swaths of image data.

For interactive systems, there exists a fourth level of parallelism to consider; the interactive level. System optimization on this level involves interleaving in parallel the man operations with the machine operations. A high rate of total throughput can be achieved if the machine can perform useful computational steps during the time that the man is thinking about his next response. In this way, the man functions as another parallel processor in the system. However, the parallel sequencing here is not as predictable or straightforward as when the microprogrammable processors interact with one another; thus there will be times when both the machine must wait for the man and the man must wait for the machine. But the major design consideration here is to allow the man to be as efficient as possible by providing the right kind of image display format and easy to use interactive tools.

The optimum display formats and interactive tools may become apparent only through system use in a real production environment. Therefore, the approach to be taken in the DMS is to provide a baseline repertoire of display modes and menus, under the assumption that future modifications and additions to the basic capability can be easily implemented.

5.2 Performance Analysis

The purpose of the DMS is to improve the production throughput which can be obtained. The unit of production to which the following analysis applies is a single model obtained from the panoramic image material.

Operations performed on the model include scanning to digitize each image of the model; non-stereo display of each image to verify quality and content, and identifying a set of points with known image coordinates for Interior Orientation; blocking of image data for data base entry; and Image Rectification. The preceding operations are all Data Preparation functions. They are performed by the full system in parallel with the Stereo Compilation functions of Image Matching, Match Point Editing, and Match Point Intersection. At this point, the results are completely defined. However, the 3-dimensional model coordinates are submitted to a final operation, Grid Normalization, performed as a Data Preparation function.

A flowchart representation of the production process is shown in Figure 5-1. Each operation identified has an estimated time associated with it. Time estimates are given in Table 5-2. The estimates are based on the complete compilation time required for a single panoramic model of 3 by 6 inch size.

Time estimates are provided for five operations performed as part of the Data Preparation function. The Image Scanning speed is chosen at the attainable rate of 10^6 pixels per second to match the Data Blocking rate attainable with a single FP. These two operations together require the time allowed for Data Blocking. The time allowance for Setup, Verification, and Interior Orientation is an estimate of operator performance instead of a firm computational requirement. The sixteen minute time includes image placement on the scanner, review of the digital image quality and coverage on the display, and interactive identification of image points for interior orientation. The Image Rectification and Grid Normalization operations are

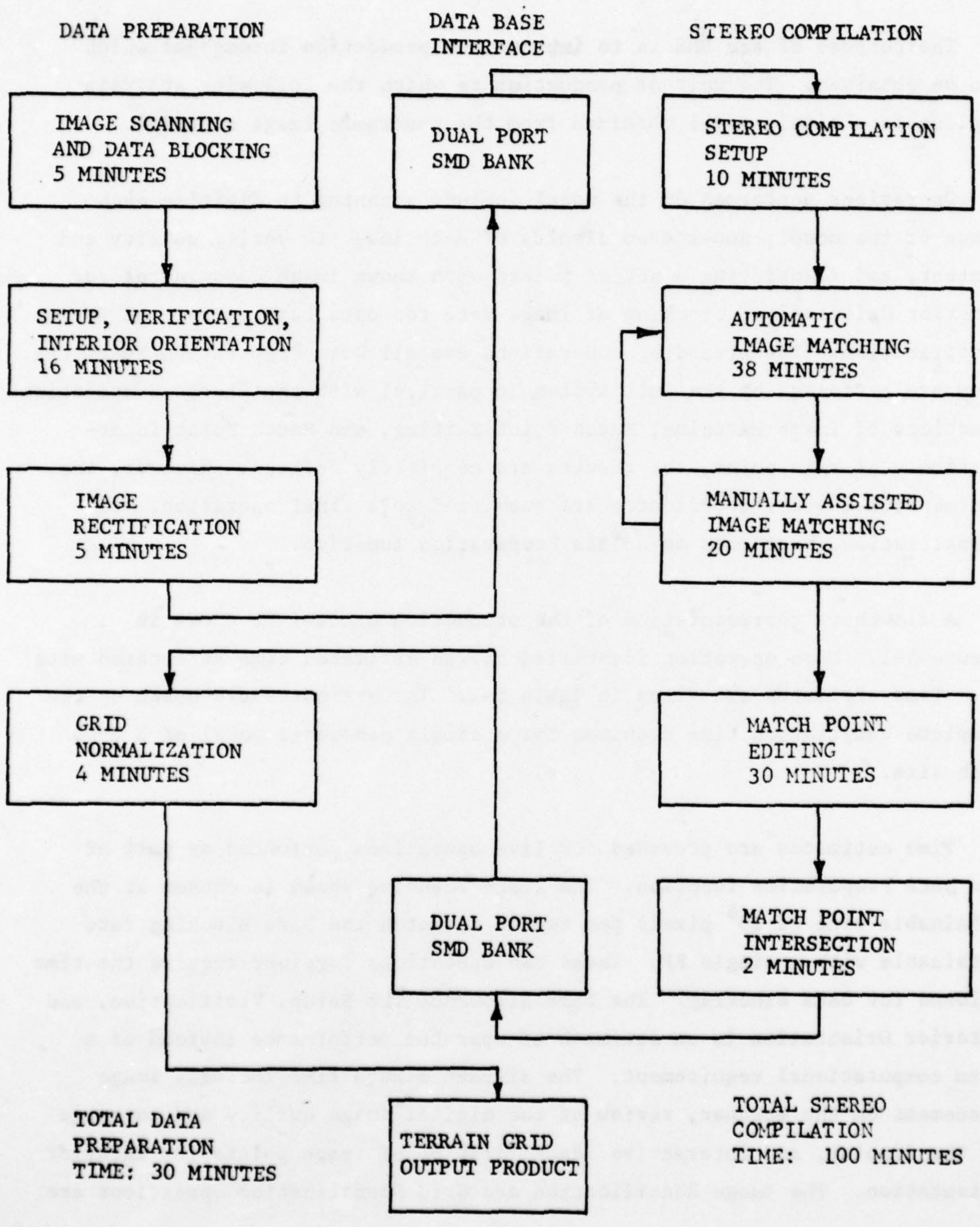


Figure 5-1. DMS Production Flow

TABLE 5-2. PRODUCTION TIME ESTIMATES

OPERATION	TIME REQUIRED	FUNCTION TYPE
Image Scanning for a Complete Model	Speed limited by Data Blocking rate of 10^6 pixels/sec	Data Preparation
Setup, Verification, and Interior Orientation of Image Scanning	16 minutes	Data Preparation
Data Blocking	5.0 minutes	Data Preparation
Image Rectification	5.0 minutes	Data Preparation
Grid Normalization	4.0 minutes	Data Preparation
Setup for Stereo Compilation	10 minutes	Stereo Compilation
Image Matching (Automatic)	38 minutes	Stereo Compilation
Image Matching (Manual Assist or Restart)	20 minutes	Stereo Compilation
Match Point Editing Including Model Inspection and Operator Defined Changes	30 minutes	Stereo Compilation
Match Point Intersection	2 minutes	Stereo Compilation

allowed 5 and 4 minutes respectively. Both of these are completely automatic operations for which the time allowances are computation time requirements. Notice that Grid Normalization is the final operation performed on the intersected match points from Stereo Compilation. It prepares the data for output in DMA standard form.

Time estimates for Stereo Compilation function operations are a similar mixture of automatic and interactive allowances, therefore, a total time estimate for Stereo Compilation is about three times as long as for Data Preparation. A single image scanner in the Data Preparation network, can support the data input requirements of two Stereo Compilation function networks. A cost-effective modification of the full production DMS Hardware Configuration would be to triple the Stereo Compilation function hardware. The single image scanner, supported by the Data Preparation network, can supply, via the Dual Port SMD Bank, the stereo model input data requirements for three independently and simultaneously operating Stereo Compilation function stations.

6.0 CONCLUSION

The primary characteristics of the Digital Mapping System design described in this report are high speed to obtain required production rates, system flexiblity, and growth capability. Fast processors operating in parallel produce the speed; and system modularity on both the hardware and software levels provides the flexibility and growth capability. The speed, however, is relative, depending on model size, the matching algorithm tuning variables, and the extent of human interaction. So too, flexibility and growth come in varying degrees. If flexibility implies the capability for procedural modifications, then both the hardware and the software in themselves are ultimately flexible. However, when the software is implemented on the hardware, the system is not quite as flexible as either component is individually; certain interfaces become frozen. The same reasoning applies to growth capability.

The DMS design at this point is highly general and still highly flexible. Enough processors have been included in the system to run the algorithms on an average model in a reasonable time frame. The memory sizes and display capabilities are sufficient to support the processing. Higher performance can be realized by replicating these components. However, a great deal of detailed design is required before the system implementation can begin.

The detailed design is contingent upon a number of tradeoffs between processing alternatives and upon unresolved parameters which directly affect the design. These alternatives and parameters can be categorized into three groups: image processing procedures, computational factors, and man-machine interaction.

In the image processing procedures category the primary area in need of further study is the scanning subsystem. Requirements must be established and analyzed regarding scanner speed, accuracy, format size, direction of scan, ease of use, and sampling interval. A panoramic image study should

also be performed, using the algorithms slated for the DMS, to determine the extent of usable pan format and the kind of algorithm tuning required for the pan materials. These factors directly affect the speed and quality of matching performance. This study was to be performed under Phase E of the effort; however, the analytical data required to process the imagery was not received until the Phase was over. A third area in need of refinement involves whether and where in the system to rectify the imagery. All the data is not yet in regarding the effects of different rectification strategies on the terrain data accuracy.

DMS design refinement in the second category, computational factors, involves the analysis of specific algorithm pieces that are critical to the accuracy and performance of the system. For example, the effect of DPCM data compression on match point accuracy is in need of analysis. Compression leads to considerable savings in memory space and improvement in throughput rate. However, if the inaccuracy induced by compression is intolerable, then compression is not a viable design consideration. Likewise, there are several analytical intersection computations of varying complexity which can be used to produce the terrain data grid. Generally, the more complex the algorithm, the more accurate the result; yet the more time-consuming the computation. Another candidate for re-evaluation in terms of DMS is the arithmetic precision required in computations used during matching in patch shape prediction and patch shaping. On a per-pixel basis, the patch shaping routine is the most time-consuming operation in the system. Its throughput can be increased either by adding processors, an expensive alternative, or by decreasing the computational precision, an inexpensive alternative. The question, then, is at what level of precision will match point accuracy become unacceptable.

Further study in the man-machine interaction category should involve the simulation and/or actual demonstration of the various DMS display modes and interactive editing options. Questions to be resolved are: what is the optimum display scroll rate; how accurately can an operator manually profile using a match point display and a stereo cursor; and how can functional menus be implemented in an easy-to-use fashion.

It is recommended that as support studies and detailed design progress, a highly collaborative effort be established between the algorithm and system designers and the end users of the system. This collaboration will ensure that design decisions have well-defined objectives and that the final DMS will meet all production requirements of the stereo mapping community.

REFERENCES

Government Documents

PS/1C/100, PS/3E/600 DMA Product Specifications for DMA Standard for Digital Terrain Elevation Data (October 1977).

PS/1CD/100, PS/1CE/100, PS/CF/100 DMA Product Specifications for Digital Landmass System (DLMS) Data Base (July 1977).

PS/3ED/601 DMA Product Specifications for Digitized Elevation Data for Firefinder (March 1978).

Non-Government Documents

Digital Cartographic Study and Benchmark: First Interim Technical Report (October 1975).

Digital Cartographic Study and Benchmark: Second Interim Technical Report (December 1975).

Digital Cartographic Study and Benchmark: Third Interim Technical Report (September 1976).

Digital Cartographic Study and Benchmark: Fourth Interim Technical Report (July 1977).